Data Mining Chemistry and Crystal Structure

Data Mining Chemistry and Crystal Structure

A DISSERTATION PRESENTED

BY

LUSANN W. YANG

TO

THE SCHOOL OF ENGINEERING AND APPLIED SCIENCES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN THE SUBJECT OF

APPLIED PHYSICS

HARVARD UNIVERSITY

CAMBRIDGE, MASSACHUSETTS

APRIL 2014

Thesis advisor: Professor Gerbrand Ceder Lusann W. Yang

Data Mining Chemistry and Crystal Structure

Abstract

The availability of large amounts of data generated by high-throughput computing and experimentation has generated interest in the application of machine learning techniques to materials science.[24] Machine learning of materials behavior requires the use of feature vectors that capture compositional or structural information influence a target property. We present methods for assessing the similarity of compositions, substructures, and crystal structures. Similarity measures are important for the classification and clustering of data points, allowing for the organization of data and the prediction of materials properties.

The similarity functions between ions, compositions, substructures and crystal structure are based upon a data-mined probability with which two ions will substitute for each other within the same structure prototype. The composition similarity is validated via the prediction of crystal structure prototypes for oxides from the Inorganic Crystal Structure Database. It performs particularly well on the quaternary oxides, predicting the correct prototype within 5 guesses 90% of the time. The sustructural similarity is validated via the prediction of Li insertion sites in the oxides; it finds all of the Li sites with less than 8 incorrect guesses 90% of the time.

Contents

Listing of figures

To my grandparents

Acknowledgments

I AM JOYFULLY INDEBTED to all the people in my life who have advised, supported, collaborated and celebrated with me throughout the course of my degree.

In particular, I am deeply grateful to my advisor Gerbrand Ceder, without whose careful and patient mentorship this thesis would not have been possible. Gerd has a deep and broad knowledge of materials science, ambitious plans for computational battery research, and a thoughtful style of mentorship that challenges his students to develop into independent scientists. I have appealed to Gerd many times for guidance and feedback over the years, and at every meeting he has encouraged and challenged me to broaden the scope of my work.

I've had the pleasure of working alongside Stephen Dacek, Shyue-Ping Ong, Geoffroy Hautier, and Bo Xu while at the Ceder group. My colleagues are friendly, talented, and hard-working, and I have enjoyed both our scientific conversations and the use of their clearly written code.

I've benefited from guidance and feedback from Alan Qi and Yao Zhu at Purdue

University, and Anthony Gamst and Randy Notestine at the University of California at San Diego. Many hours of conversation with Alan, Yao, Anthony and Randy were formative to my education in the field of machine learning.

My time at MIT has been highlighted by the cheerful and steadfast support of my officemates, past and present; ShinYoung Kang, Ian Matts, Rahul Malik, Nancy Twu, Ruoshi Sun, and Yabi Wu. May we enjoy many meals together in the years to come!

My time in Boston would not have been complete without many wonderful adventures, from art fairs to apple picking, dancing to rock climbing. Looking back, I find that I appreciate most of all the friendships I've forged with Raji Shankar, Emily Eames, Jasmine Eleftherakis, and Shawn Ligocki, and my housemates Emily and Kirby Russell, Dale Winter, Alexey Spiridonov, Naomi Mackenzie, Alan Morse, and Tom Dimiduk.

Finally, I am grateful for the love and support of my parents and my brother. Many years of effort on their part have instilled within me the belief that math is fun and research is worthwhile.

0 Introduction

Growing materials databases, computational power, and the availability of better computational techniques have made it an exciting time in the computational study of materials science. The Materials Project has published *ab initio* computations of almost 50,000 compounds online, including over 20,000 band structures.[42] The Inorganic Crystal Structure Database now contains 161,030 entries.[7] Growing databases of materials incur the necessity to develop methods with which to organize such knowledge, and allow for the possibility of systematically mining this data for patterns.

This thesis is about quantifying, organizing, and extracting patterns from data. This thesis provides a framework for analysis of structure, quantifying chemistry, the analysis of substructures and void spaces within crystal structures, and the prediction of new structures. The work in this thesis creates similarity functions between ions, compositions, substructures and crystal structures, quantifying the relationships between ions, compositions, substructures, and crystal structures. This thesis draws heavily upon

historic, heuristic methods of understanding crystal structure to build a quantitative framework by which we can use modern computational tools to analyze modern materials databases.

We begin chapter 1 with a review of crystal structure prediction in which we cover historic and modern methods of structure prediction. In chapter 2, we present a data mined similarity function between ions that quantifies how likely ions are to substitute for each other within the same crystal structure. In chapter 3, we extend the ionic similarity function to define a similarity function between compositions, and show that compounds with similar compositions are likely to have similar crystal structures. In chapter 4, we define a substructure similarity function between atomic neighborhoods that quantifies chemical and geometric similarity. We use the substructure similarity function to search for Li ion insertion sites for Li ion battery cathodes. Finally, in chapter 5, we conclude with a few thoughts on future applications of this work.

1

Structure Prediction

"One of the continuing scandals in the physical sciences is that it remains impossible to predict the structure of even the simplest crystalline solids from a knowledge of their composition"[53], said John Maddox in his 1988 Nature publication entitled 'Crystals from first principles.' The crystal structure of a material is a critical piece of information from which many material properties may be computed prior to synthesis. Indeed, large materials databases use crystal structure information to perform high throughput computations on thousands of materials.[42]

The websites that host such large databases regularly compute properties that include band gap, formation energy and magnetic moment, taking advantage of recent advances in computational power to search for new materials. Application-specific *ab initio* calculations include voltage, diffusion, and catalysis activity computations for Li-ion battery cathodes,[3,86,100,29], Li-air batteries,[46] water splitting photocatalysis for solar energy,[99] and super ionic conductors[63].

While computational screening of thousands of materials has the power to greatly accelerate materials design and discovery, computational methods rely heavily on the knowledge of the crystal structure of the material at hand. For as yet unsynthesized materials, such knowledge is not readily available. As Woodley and Catlow write in their 2008 review paper, "structure prediction is not ... yet routine."[98] In this chapter, we will present a brief overview of the state of crystal structure prediction. We begin with an overview of traditional methods of structure and substructure prediction and analysis, continue on to survey modern methods, and conclude with a discussion of data-mining techniques.

1.1 Traditional Methods

Historically, crystal structure prediction has been understood using a series of simple, efficient rules based upon physical insights. These heuristic methods rely upon physical properties of the ions at hand, including ionic radius, charge state, and Mendeleev number. A learned scientist searching for new materials would combine his experience with known materials with his knowledge of these heuristics to make educated guesses as to the structure of a new compound.

The Pauling Rules[75,76] for the structure of complex ionic crystals, published in 1929, are perhaps the most well-known heuristic rules for crystal structure prediction. Pauling describes the local environments of each cation, creating polyhedral building blocks which he then packs into coherent crystal structures. Pauling begins by creating coordinated polyhedra of anions about each cation. The ratio of cation radius to anion radius determines both the cation-anion distance and the coordination of the central cation; this rule is guided by the physical intuition that anions and cations attract, packing as many anions about each cation as space allows. Pauling uses similar electrostatic arguments to limit the number of polyhedra that share corners, edges, and faces. Finally, Pauling ends with the thought-provoking statement that "the number of essentially different kinds of constituents in a crystal will be small."[75] The Pauling rules take into account both geometric, space-filling concepts and chemical concepts, using simple electrostatic arguments to support geometric packing rules.

I.D. Brown published in 1981 the bond valence method, an extension of the Pauling rules based on chemistry rather than geometry.[8] Each ion in a crystal structure has a va-

lence equal to the number of electrons the ion uses for bonding. Additionally, Brown assigns each *bond* in the crystal structure a valence equal to the number of electron pairs forming the bond. The bond valence method is used to verify experimentally determined crystal structures and to assess and improve the quality of randomly generated trial structures.[9]

One of the key concepts behind heuristic methods is that patterns arise when organizing crystal structure information via any number of physically motivated parameters. Such structural information is often viewed in the form of a *structure map*, where compounds are arranged against two axes based upon parameters such as ionic radius mismatch, valence electron concentration, or electronegativity difference. Mooser and Pearson realized in 1959 that organizing compounds by electronegativity difference, ionicity, and average principle quantum number clusters together compounds of the same crystal structure.[66] Structural trends can also be found when organizing compounds via parameters derived from pseudopotentials.[32,38,84,43] . Villars[87,88,89] and Muller and Roy[68] have created a number of structure maps based upon the parameters mentioned above. Finally, Pettifor, frustrated by the limitations of the various parameters described above - which have varying degrees of success in separating crystal structures across different chemistries - created a phenomenological scale X that maps each element to a single number along that scale.[78,77] Using this phenomenological scale, Pettifor created structure maps for multiple binary systems, achieving good structural separation. A sample Pettifor map is shown in Figure 1.1.

While the heuristic methods outlined above form the theoretical backdrop for this thesis, their weakness lies in their lack of predictive capability. These methods are qualitative in nature, and it is unclear how to extrapolate the patterns found to predict the structures of ternary or quaternary compounds.[67]The work in this thesis does not map ions to a phenomenological scale X, but instead extracts a quantitative similarity function between ions. This quantitative similarity function does not rely upon Mendeleev number, ionic radius, nor electronegativity; instead, it is mined directly from structurally similar compounds. The ionic similarity function is extrapolated into a quantitative similarity function between compositions, allowing for a systematic extension of Pettifor's structure maps that clusters together materials with similar crystal structures across all stoichiometries.

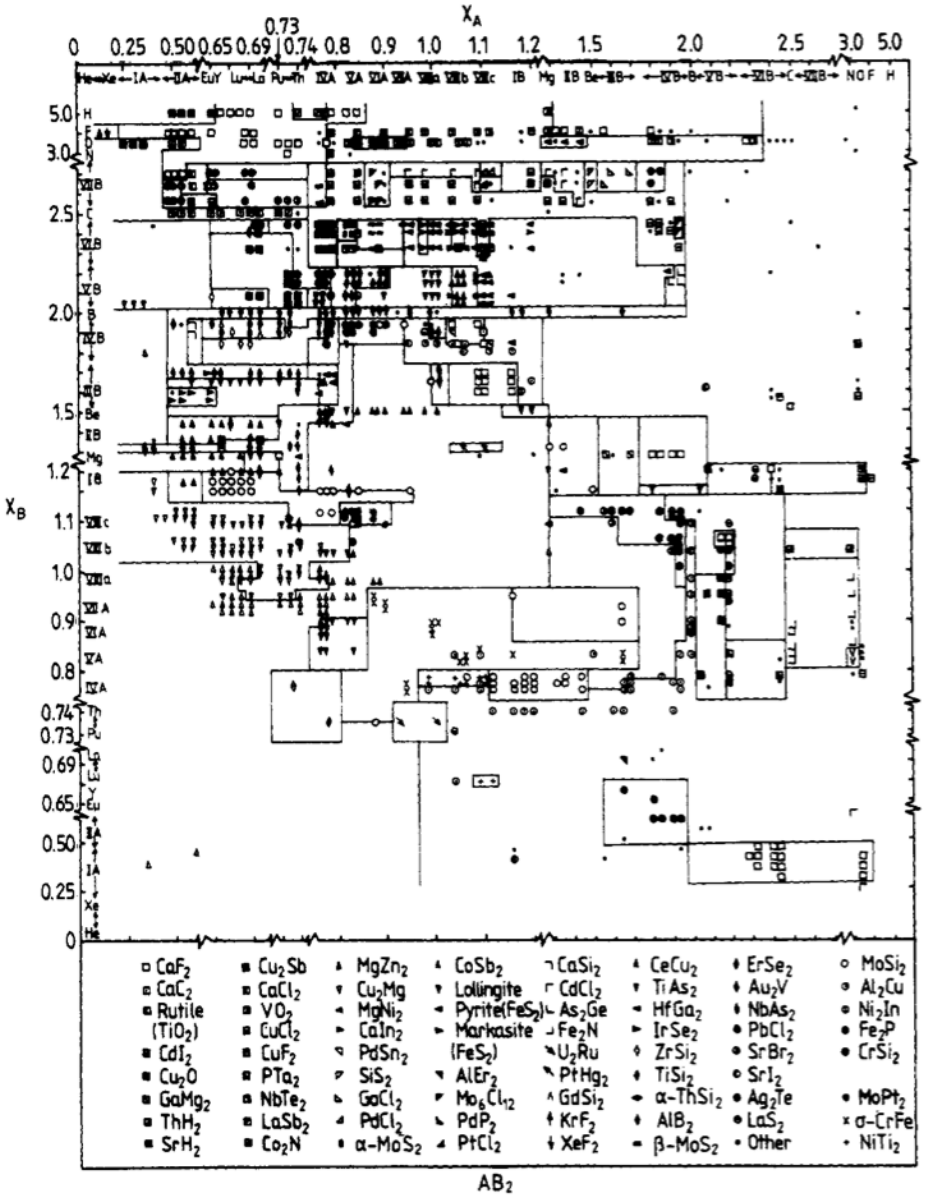


Figure 1.1: Pettifor's structure map for AB_2 compounds.[78] Plotting compounds via Pettifor's phenomenological scale X allows for good separation between structure prototypes.

1.2 Substructure

Substructural analysis, or analysis based upon atomic environments or coordination polyhedra, also has a rich history in materials science. While the Pauling rules began with coordination polyhedra in ionic crystals, Laves and Witte published in 1935 a set of arguments pertaining to the crystal structures of metallic alloys, modeling them as dense packings of hard spheres with differing radii. When the sizes of the metallic atoms are favorably mismatched, certain crystal structures can be favored. Frank and Kasper followed their work in 1958[25] with a geometric theory of topologically close-packed phases. They begin by putting forth geometric arguments for a few commonly found coordination polyhedra, and then discuss the geometric constraints imposed by tiling space with these coordination polyhedra.

More recently, Daams and Villars have published an impressive survey of the atomic environments in over 200,000 inorganic compounds.[21,20] They define an 'atomic environment type', a geometric construct that describes the positions of neighbors around a central ion. They find that 80% of the crystal structures investigated can be described with only the 20 most commonly found atomic environment types, whereas the remaining 20% of compounds requires adding 50-70 rarely occurring atomic environments. The twenty most commonly found atomic environment types, shown in figure 1.2 are often highly symmetric.

The physical and geometric arguments given by Pauling, combined with the survey by Daams and Villars, support Pauling's original statement that "the number of essentially different kinds of constituents in a crystal will be small."[75] We have reason to believe that the number of different crystal substructures might not be intractably large, which would yield a data set too sparse to data mine. For example, if there were millions of differently shaped substructures, it would be difficult to find useful correlations in a data set of only 5,000 crystal structures. In chapter 4, we explore the possibility of data mining crystal substructures.

1.3 Coordinate Search

Maddox's 1988 paper on crystals from first principles states, "one would have thought that, by now, it should be possible to equip a sufficiently large computer with a suf-

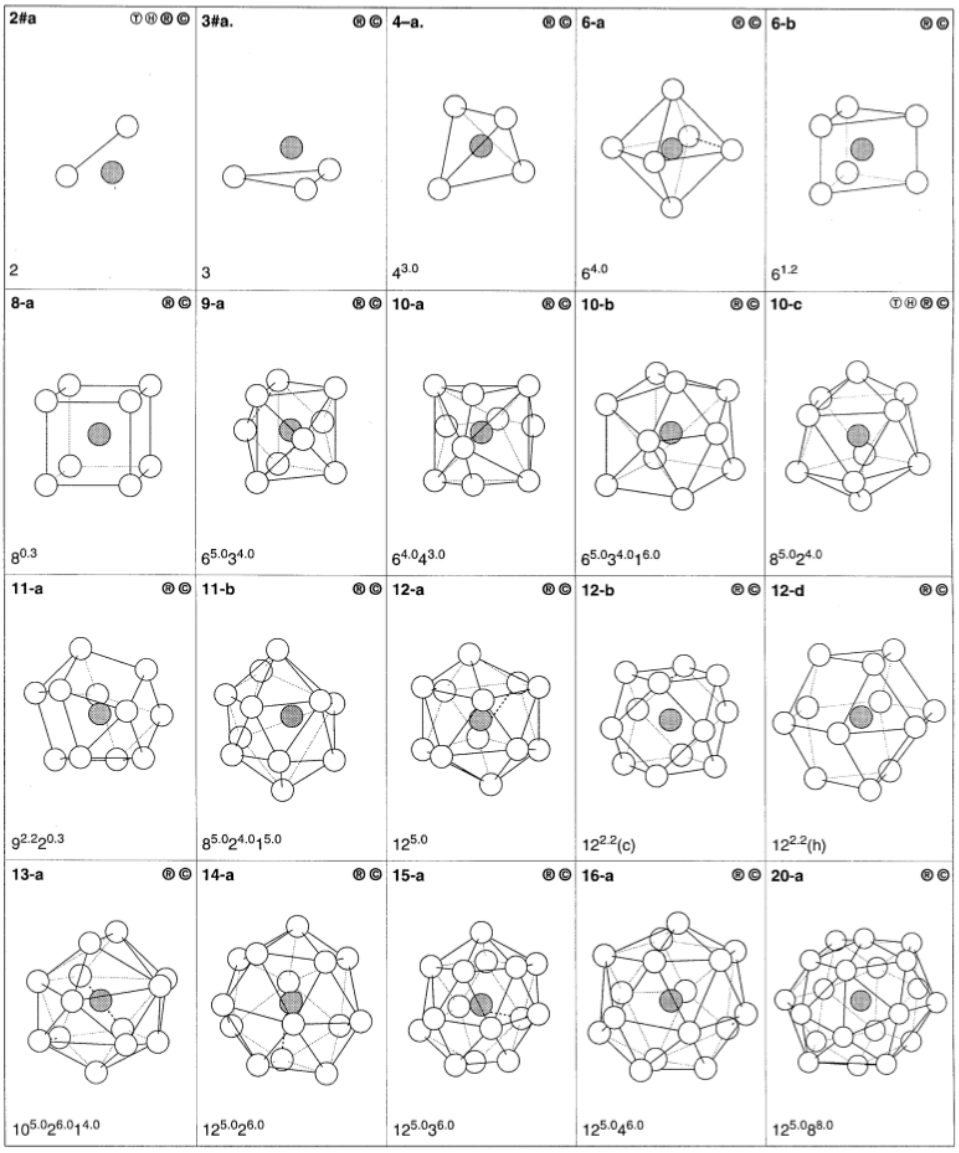


Figure 1.2: Daams and Villars' 20 most common atomic environment types that account for 80% of the inorganic crystal structures investigated.

ficiently large program, type in the formula of the chemical and obtain, as output, the atomic coordinates of the atoms in a unit cell."[53] Computationally solving for the ground state crystal structure from first principles can be seen as a minimization problem; search through the space of possible crystal structures for the lowest energy structure. Armed with density functional theory, the energy of a crystal structure may be accurately evaluated to within well-understood limitations.[13,30,54,55,2,79] However, given that the size of the unit cell necessary to describe a crystal structure is unknown, the search space is infinite. Even given the size of the unit cell, the size of the search space is immense and has many local minima.[70,82,72] In this section, we give a short overview of two search algorithms that intelligently sample the search space.

Simulated annealing[48] is a search method that mimics what happens in nature as a metal cools from a liquid state. Beginning with a disordered state, random perturbations are suggested, then accepted or rejected via a Monte Carlo algorithm. Perturbations which increase the energy are accepted with low probability, whereas perturbations which decrease the energy are accepted more frequently. As the simulation progresses, the temperature is lowered, decreasing the probability with which energetically unfavorable perturbations are accepted. If the annealing process is carried out slowly enough, the ground state solution is probabilistically favored.

Simulated annealing has been used to validate experimental structures[74], to predict the crystal structures of a few simple inorganic solids[83,81], and to predict the structures of several biomolecules.[93] Simulated annealing can take a long time to converge because the temperature must be lowered slowly, requiring hundreds or thousands of energy evaluations. Another limitation of simulated annealing is that each run must begin with a single point, which limits the breadth of the search space and it is thus possible that not all low-energy areas will be found.[98,92]

Mellot-Draznieks et al. have published several interesting studies using simulated annealing techniques to assemble "secondary building units."[59,58,57,56] These secondary building units are predefined inorganic building units in three-dimensional spaces. Mellot-Draznieks uses empirical "glueing" rules[58] to scan through potential packings of these building units. Dyer et al. expanded upon this work in 2013, using larger modules to investigate perovskite-related materials.[22,14] While the assembly methods used by Draznieks and Dyer are extraordinarily promising, these structure prediction techniques depend upon the user's chemical intuition to select plausible secondary building

units or modules. The work in this thesis provides a framework for the systematic, quantifiable identification of likely building blocks from existing data.

Genetic algorithms[17] search through configuration space by mimicking not the cooling of metal from a liquid state, but rather Darwinian evolution. An initial population of crystal structures is mated or mutated via a series of random operations that include splicing, swapping ions, and random ionic displacements. After mutation, the population is again evaluated, and the 'fittest' - i.e. the structures that are lowest in energy - survive. The process is repeated. Over time, structures with lower energy features are more likely to survive.

Genetic algorithms have many variations in implementation. Crossover routines vary how to splice and join components from different crystal structures. Mutation routines vary as well, and the rate of crossover versus mutation is variable. Furthermore, choosing which structural candidates survive for the next round has several different implementations. For example, the next population can consist of only the best candidates from the previous mutation, or can also include some sub-optimal candidates.[17,50,33,44,96,1,102]

Genetic algorithms have been successfully applied to metallic and alloy clusters, microporous oxide structures, and molecular crystal,[44,96,101] among many other applications. Two notable successes of the genetic algorithm approach include the prediction of previously unknown crystal structures Li_3RuO_4 and a new phase of high-pressure boron.[12,69] The weaknesses of genetic algorithms lie in their significant computational cost. Probert et al.[1] required more than 40 local optimizations to find the structure of bulk silicon; Glass et al.[28] required 390 energy relaxations to find the structure of $MgSiO_2$. Additionally, the many parameters that describe a particular genetic algorithm are often varied from system to system; genetic algorithms are a powerful tool that benefit greatly from user experience.

1.4 Data Mining

Growing materials databases and improving computational power in the era of 'big data' have prompted the rise of a new method of structure prediction. Instead of relying upon either the human execution of a set of qualitative, heuristic rules, or *ab initio* calculations sampling configuration space, data mining approaches capture previously

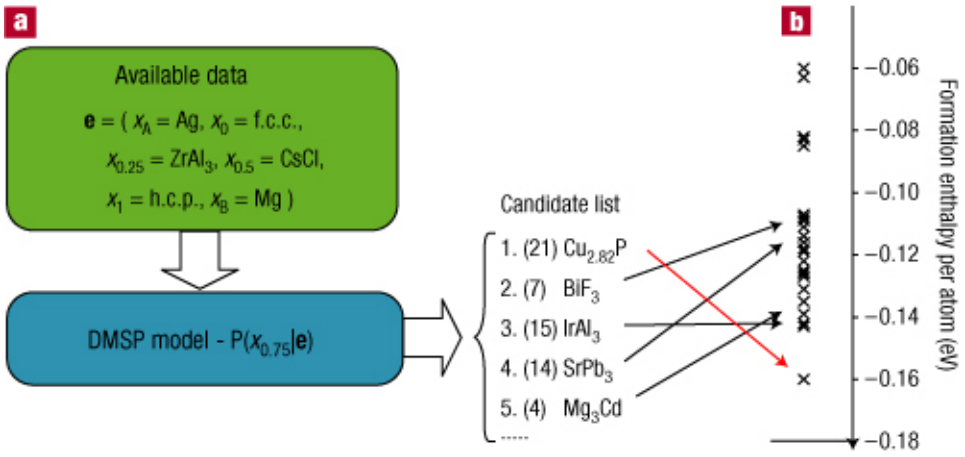


Figure 1.3: Fischer's data mined structure prediction algorithm for $AgMg_3$.[24] a) Given available data from the database (green box), Fischer's algorithm ranks candidate crystal structures by the probability with which $AgMg_3$ forms in them. b) *Ab initio* formation energy for the top 5 candidate crystal structures along with 26 additional structure types.

heuristic knowledge quantitatively. Curtarolo et al.[19] performed principal component analysis on a set of 55 binary metallic alloys in 2003, calculating their energies over a set of 114 different crystal structures and showing that the energies of differing crystal structures were strongly correlated between different chemical systems.

Fischer et al. built upon this work by mining structural correlations from binary crystal structures in the Pauling File.[24,23,90] In an extraordinarily ambitious paper, Fischer attempts to extract for all binary metallic systems, for all compositions, for each crystal structure, the probability that a compound forms. Figure 1.3 depicts a schematic of the prediction algorithm for the Ag-Mg system. Taking in the Pauling File, information on the two compounds at hand, and the stoichiometries and crystal structures at which they are known to form compounds, Fischer produces a probability that a given compound will form in this system.

Over the binary systems investigated, Fischer had a 90% chance of guessing the correct crystal structure for a compound that forms within the first 5 guesses. Figure 1.4 shows aggregated performance data for Fischer's algorithm. The blue line shows how many guesses are necessary to find the correct crystal structure via Data Mined Structure Prediction versus probability; the green line shows the analogous result for random structure guessing, and the red line shows the analogous result for guessing struc-

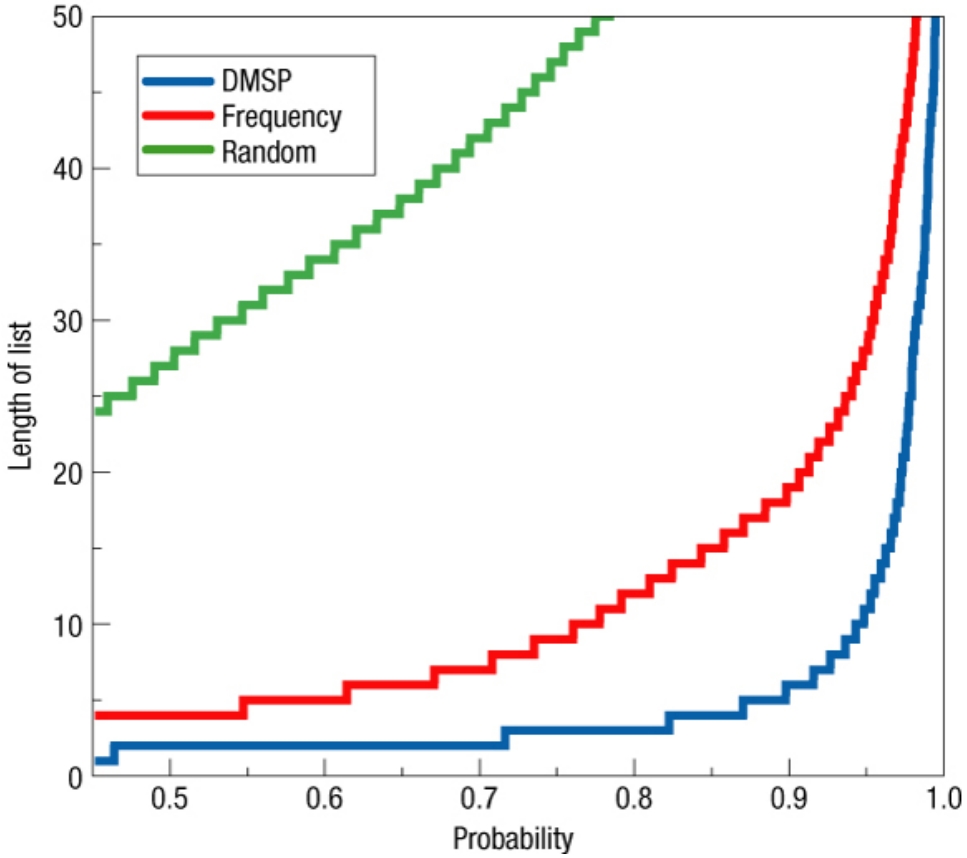


Figure 1.4: Performance of Fischer's Structure Prediction Algorithm.[24] The length of the candidate list required to find the true crystal structure with a certain probability with DMSP (Data Mined Structure Prediction), random structure guessing, or suggesting structures on the basis of the frequency with which they appear.

tures based upon the frequency with which they appear.

Fischer's pioneering work is impressive in scope. Unlike many structure prediction algorithms, Fischer predicts not only the crystal structure but also whether or not a compound will form. Additionally, the number of energy evaluations necessary to find the ground state crystal structure is typically very low. However, Fischer's algorithm suffers from two weaknesses. Firstly, as a structure prediction algorithm, it functions well on binary alloys but is not easily extended to to more complex structures. Fischer's algorithm suffers from data sparsity. As the complexity of the system at hand grows, more data is necessary to fit the parameters of the cumulant expansion. Secondly, the

crystal structures Fischer suggests are limited to the crystal structure prototypes that have already been seen in the database at hand; Fischer cannot construct novel crystal structures.

More recently, Hautier et al. have expanded upon Fischer's work, applying the cumulant expansion technique to the ternary oxides.[37,36,35] Figure 1.5 depicts the quality of Hautier's results on the ternary oxides. The length of the crystal structure candidate list needed to select the correct crystal structure is plotted versus the probability that that the correct structure is in the candidate list for three different values of $p_{compound}$. A prediction with a low threshold for $p_{compound}$, shown in the blue line, recovers 100% of the ternary oxides in the Inorganic Crystal Structure Database 2012; this prediction performs more poorly, requiring more than 20 guesses to identify the correct crystal structure at 95% accuracy. Predictions with higher thresholds for $p_{compound}$ recover smaller percentages of the ternary oxides in the ICSD but predict crystal structures with more confidence (shown in green and red). The accuracy with which the algorithm predicts the correct crystal structure decreases as the percentage of recovered ICSD oxides increases.

In his next paper, Hautier laid the groundwork for this thesis by data mining ionic substitutions. Prompted by the data sparsity exhibited in the ternary oxides, Hautier's work searches for ionic substitutions that occur in binary and ternary oxides and uses the commonly found substitutions to predict substitutions in quaternary compounds. Hautier's ionic substitution model is the starting point for this thesis, and is expanded upon further in Chapter 2.

More recently, Rupp et al. have built a fast and promising model that quickly and accurately predicts the atomisation energies of molecular crystals.[80] Rupp extracted a symmetric Coulomb matrix representation from molecules, shown in figure 5.2. Rupp feeds the Coulomb matrix representations of molecules into a kernel ridge regression model via a distance function defined as the Euclidean norm of their diagonalized Coulomb matrices: $d(M, M') = d(\epsilon, \epsilon') = \sqrt{\sum_I |\epsilon_I - \epsilon'_I|^2}$, where ϵ is the eigenvalue of the Coulomb matrix M. Rupp achieves extraordinarily good results on the atomization energies of small organic molecules with up to seven "heavy" atoms that contain C, N, O, or S and are saturated with hydrogen atoms. Rupp's methods are currently being expanded and improved upon, and have recently been extended to learn electronic transport properties of one-dimensional nanostructures.[31,51,64,65]

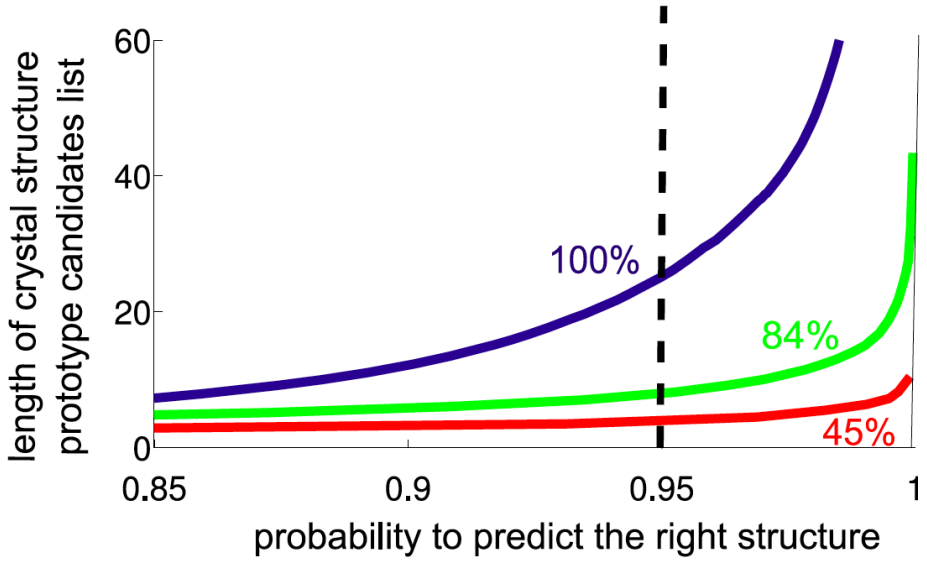


Figure 1.5: Performance of Hautier's structure prediction algorithm on the Ternary Oxides.[35] The length of the candidate list required to find the true crystal structure versus the probability that the correct crystal structure is in the candidate list for three different values of $p_{compound}$. Lower values of $p_{compound}$ recover higher percentages of the ternary oxides in the Inorganic Crystal Structure Database (percentages labeled), but require more guesses to select the correct crystal structure.

This thesis provides a data mining framework for the study and prediction of crystal structures. In chapter 2, a reconstruction and extension of Hautier's ionic substitution model is presented. Strongly motivated by D.G Pettifor's structure maps[78,77] we present in chapter 3 a similarity function between two compositions that preserves correlations between composition and crystal structure. Drawing heavily upon the ideas of Pauling, Laves and Witte, Frank and Kasper, and Daams and Villars,we present in chapter 4 a definition of crystal substructure and a similarity function between substructures that reflects geometric and chemical similarity. Finally, in chapter 5 we discuss limitations and potential extensions of this work.

2

Ionic Substitution Similarity

The overarching goal of this thesis is to develop quantitative similarity functions between compositions, substructures, and entire crystal structures that respect both chemical and geometric similarity. Therefore we begin with a definition of chemical similarity.

A chemical similarity function $\mathrm{Sim}_{\mathrm{ion}}(i_1, i_2)$ should take two ions i_1 and i_2 and returns a number that is higher if those two ions are similar in a manner that is pertinent to the data mining problem at hand. Such a function should also satisfy

- $0 \leq \mathrm{Sim}_{\mathrm{ion}}(i_1, i_2) \leq 1$, and
- $\mathrm{Sim}_{\mathrm{ion}}(i_1, i_2) = 1$ if $i_1 = i_2$

With respect to this thesis, the problem at hand is the prediction of crystal structure. Therefore, the similarity function must also satisfy

- $\mathrm{Sim}_{\mathrm{ion}}(i_1, i_2)$ should grow monotonically with the probability that i_1 and i_2 substitute for each other within a given prototype.

We use a similarity function developed by Hautier et al[36] based upon the frequency with which two ions will substitute for each other within the same structure prototype. Hautier's work allows a function derived from experimental data to quantify a scientist's understanding of ionic similarity. Like the periodic table, the similarity function imparts structure upon the space of ionic species.

In this chapter, we present a reconstruction of this work. We begin by defining the terminology used throughout this thesis, including what it means for two crystal structures to share the same prototype. We describe the model by Hautier that forms the basis for the $\mathrm{Sim}_{\mathrm{ion}}$ function, the algorithm used to extract $\mathrm{Sim}_{\mathrm{ion}}$ from a given database, and we discuss several parameters involved in tuning $\mathrm{Sim}_{\mathrm{ion}}$ for usage in specific applications. We discuss the use of similarity functions in general the $\mathrm{Sim}_{\mathrm{ion}}$ function in particular, with respect to clustering and imposing a structure upon the space of ions. Finally, we end with a few thoughts on possible extensions of this work.

2.1 Terminology

We begin with some terminology that will be used throughout this thesis.

2.1.1 Definitions

The following definitions are widely used within the materials science community.

- A *species* is a specific chemical identity consisting of an element and its charge state.
- A *composition* is a set of species and the ratios in which they appear.
- The *Bravais lattice* of a crystal structure is given by three fundamental translation vectors a_1, a_2, and a_3. These lattice vectors must be such that the structure is identical when translated by any vector $r' = r + n_1a_1 + n_2a_2 + n_3a_3$, where n_1, n_2, and n_3 are all integers.

- A *basis* is a description of the arrangement of ions within a particular unit cell of a crystal structure. For each site within a basis a species is given as well as the location within the unit cell.
- A *crystal structure* is an infinitely periodic, 3-dimensional arrangement of atoms in space. A crystal structure can be described via a basis and a Bravais lattice. By definition, the information given in a crystal structure includes its composition.

While the following terms are also commonly used within the scientific community, their usage is not always clearly defined. For the sake of clarity, the following terms are defined within the scope of this thesis:

- An *affine mapping* between crystal structures is one that preserves points, straight lines, and planes. Affine transformations include translation, scaling, reflection, rotation, and any combination of the above.
- The *chemical equivalence class* of a given site within a crystal structure is the set of all sites of that crystal structure with the same species.
- A *prototype* describes a family of crystal structures that can be transformed into each other via affine mappings that preserve chemical equivalence classes.
- A *compound* describes a particular, unique entry in a materials database. Two entries with the same crystal structure are considered duplicates, and are thus the same compound.

2.1.2 An Example

We illustrate the usage of these terms on the CsCl and Fe crystal structures, shown in figure 2.1.

- The *species* that make up CsCl are Cs^+ and Cl^-.
- The *composition* of CsCl is $\{Cs^+ : 1, Cl^- : 1\}$.
- The *Bravais lattice* of CsCl is cubic. $a_1 = l(0, 0, 1)$, $a_1 = l(0, 1, 0)$, $a_1 = l(1, 0, 0)$, where the lattice constant $l = 4.123$Å. The Bravais lattice

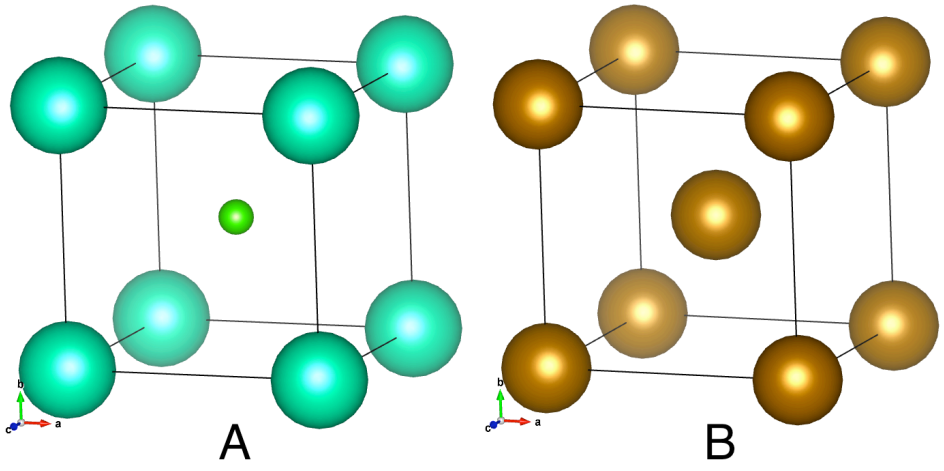


Figure 2.1: The CsCl crystal structure is shown in figure A. The Fe crystal structure is shown in figure B. Although the two structures can be mapped onto each other via an affine mapping, this mapping does not preserve chemical equivalence class, therefore these two structures do not share the same structure prototype.

of Fe is that of a body-centered cubic lattice, which can either be given as a cubic conventional unit cell $a_1 = l(0,0,1), a_1 = l(0,1,0), a_1 = l(1,0,0)$, or as primitive unit cell: $a_1 = \frac{l}{2}(1,1,-1), a_1 = \frac{l}{2}(1,-1,1), a_1 = \frac{l}{2}(-1,1,1)$. $l = 2.87$Åfor both cases.

- The *basis* accompanying the cubic CsCl crystal structure is given by {Cs^+ at (o, o, o), Cl^- at $(\frac{1}{2},\frac{1}{2},\frac{1}{2})$ }. The basis accompanying the conventional unit cell of Fe requires a two ion description, given by {Fe at $(0,0,0)$, Fe at $(\frac{1}{2},\frac{1}{2},\frac{1}{2})$}; however, the primitive unit cell of Fe requires only a one-ion basis: {Fe at $(0,0,0)$}.

- Although the two lattice constants for CsCl and Fe are not equal, it is possible to map all the ions of one crystal structure onto the ions of the other crystal structure via an *affine mapping*.

- The *chemical equivalence class* of Cs^+ in the CsCl crystal structure is the set of all Cs^+ ions; similarily, the equivalence class of Cl^- is the set of all Cl^- in that

crystal structure. In this simple example, the chemical equivalence class of Fe in the Fe crystal structure is also the set of all Fe atoms;

- Lastly, because it is impossible to map the equivalence class of Fe onto either the equivalence class of Cs^+ or Cl^- ions, Fe does not have the same *prototype* as CsCl.

2.2 Modeling Ionic Substitution

The ionic substitution similarity model is based upon an algorithm that counts the number of times one ion substitutes for another within the same prototype within a given database. In this section, we derive the mathematical model that is used to compute the similarity function, and we illustrate the substitution counting process with a few examples.

2.2.1 Derivation

Hautier et al[36] model the probability of two crystal structures, X and X', taking the same prototype as a function of the ion-ion substitutions $i = (i_1, i_2)$ required to map the chemical equivalence classes of X onto those of X', given that an affine mapping exists that will map the locations of the ions onto each other as appropriate.

$$p(\text{prototype}(X) = \text{prototype}(X')) = \frac{e^{\sum_i \lambda_i f_i(X, X')}}{Z} \tag{2.1}$$

We recognize the form of the equation from statistical mechanics, where the probability of a particular state is inversely proportional to the exponential of its energy.The $f_i(X, X')$ are a series of binary indicator functions, given by

$$f_i(X, X') = \begin{cases} 1, & \text{if ion } i_1 \text{ substitutes for ion } i_2 \\ 0, & \text{else} \end{cases} \tag{2.2}$$

The sum is taken over all pairs of ionic substitutions $i = (i_1, i_2)$. The $f_i(X, X')$ indicate whether or not a specific binary substitution i occurs between two crystal structures X and X'; if the binary substitution occurs, the indicator function evaluates to 1 and the probability of this occurrence is weighted by λ_i. Otherwise, indicator function

evaluates to 0 and the term corresponding to the substitution i is neglected. Returning to our analogy with statistical mechanics, λ_i can be thought of as the energetic bonus or penalty associated with the substitution i; commonly found substitutions will have a larger bonus, whereas rarer substitutions will have a more negative bonus.

Finally, Z is a partition function which normalizes $P(X, X')$ such that the sum of all the probabilities is 1. Assuming that each binary substitution i is independent of the others, we have

$$Z = \prod_i (e^{\lambda_i} + 1) \tag{2.3}$$

2.2.2 Examples

Counting substitutions has a number of nuances, which we will illustrate here with a few examples:

Example 1: A Simple Example

Suppose the database at hand consisted of two prototypes. The first prototype contains the compounds SiO_2 and GeO_2, and the second prototype contains the compound TiO_2. The three crystal structures are shown in figure 2.2.

In this example, Si^{4+} substitutes for Ge^{4+} in the first prototype. We do not count substitutions of one O^{2-} ion for itself, as self-similarity is defined to be 1. Lastly, the third crystal structure TiO_2 cannot contribute to a substitution count, as it is the only structure in its prototype.

Example 2: Concurrent substitutions

Suppose a prototype with two compounds contained Na_2MnO_4 and Ca_2SiO_4 as shown in figure 2.3. In this example we illustrate that multiple substitutions can occur concurrently. In this case the substitution count would be: { (Na^+, Ca^{2+}) : 1, (Si^{4+}, Mn^{6+}) : 1 }. , Not only does this model allow for multiple substitutions, but it also allows for charge-imbalanced substitutions.

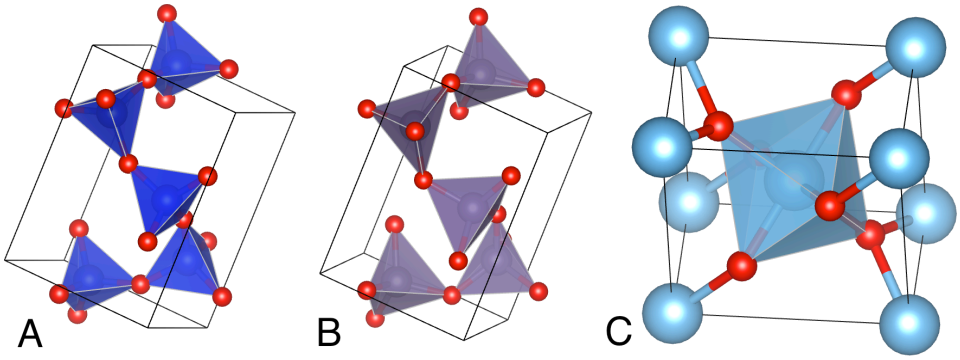


Figure 2.2: The SiO_2 and GeO_2 crystal structures are shown in figures A and B; they share the same prototype. The TiO_2 crystal structure is shown in figure C.

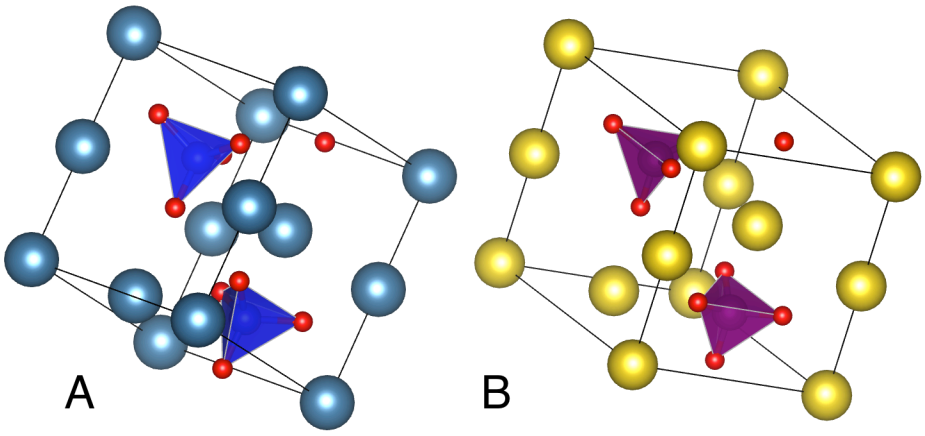


Figure 2.3: The Na_2MnO_4 and Ca_2SiO_4 crystal structures are shown in figures A and B; they share the same prototype. Note that multiple substitutions can occur simultaneously, which allows for charge-imbalanced substitutions to occur.

Table 2.1: Substitution Count for The Prototype Including Na_2CrO_4

i_1	i_2	count
Na^+	Li^+	3
Cr^{6+}	S^{6+}	2
Cr^{6+}	Fe^{6+}	1
S^{6+}	Fe^{6+}	2

Example 3: Counting Substitutions

Consider the case of a prototype with 4 compounds: Na_2CrO_4, Na_2FeO_4, Li_2SO_4, and Na_2SO_4. The four corresponding crystal structures are shown in figure 2.4, and the substitution counts are given in table 2.1.

The substitution count for a given ion-ion pair i is given by the number of pairs of compounds within the same prototype in which i occurs; although there are only 4 compounds in this example prototype, we record 5 possible substitutions for the Cr^{6+} equivalence class. There are 2 substitutions of Cr^{6+} for S^{6+}, 2 substitutions of Fe^{6+} for S^{6+}, and one substitution of Cr^{6+} for Fe^{6+}. Indeed, the number of substitutions that a prototype contributes to the count can grow very quickly with the number of compounds in the prototype. In a worst-case scenario, a prototype with n compounds could contribute $\mathcal{O}(n^2)$ substitution counts per chemical equivalence class. For this reason, we expect the substitution count to be dominated by the most common chemistries occurring in the largest prototypes.

2.3 Extracting Ionic Substitution Similarity from Data

Given the model defined above, we use a common algorithm in the data mining community called *maximum likelihood*; we extract the model parameters λ_i by maximizing the probability of the database at hand occurring. The data sets used in this thesis are described in Appendix ??. The probability of all the points in a data set D occurring is given by the joint probability of each pair of compounds (X, X') occurring in the same prototype:

$$p(\{(X, X') \in D\}) = \prod_{(X,X')\in D} \frac{e^{\sum_i \lambda_i f_i(X,X')}}{Z} \tag{2.4}$$

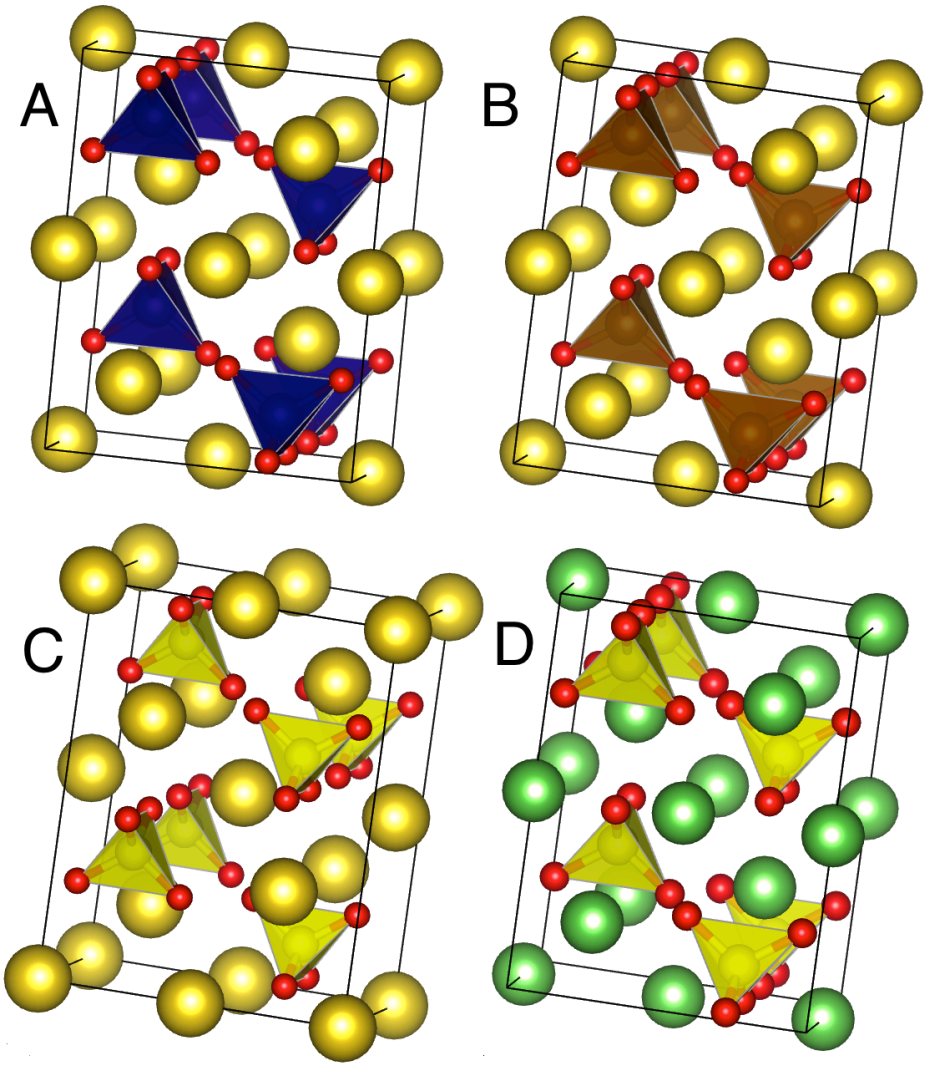


Figure 2.4: Four compounds in the same prototype are shown. Na_2CrO_4 is shown in figure A; the small red ions are O^{2-}, the large yellow ions are Na^{+}, and the blue tetrahedra have Cr^{6+} centers. In figures B and C, those tetrahedra have Fe^{6+} and S^{6+} centers. Figure D shows S^{6+} - centered tetrahedra in a framework of green Li^{+} cations. Counting substitutions in this prototype yields two instances of S^{6+} substituting for Cr^{6+} and Fe^{6+}, one instance of Cr^{6+} substituting for Fe^{6+}, and three instances of Na^{+} substituting for Li^{+}.

A commonly used algebraic trick is to maximize the probability by taking the logarithm of both sides. Assuming that each binary substitution i is independent of the others, and thus the λ_i are independent, we take the derivative of the log likelihood and set it equal to zero:

$$\begin{aligned}
\log P(\{(X, X') \in D\}) &= \sum_{(X,X')\in D} \left[\sum_i \lambda_i f_i(X, X') - \log Z \right] \\
\frac{\partial \log P(\{(X, X') \in D\})}{\partial \lambda_i} &= \sum_{(X,X')\in D} \left[\lambda_i f_i(X, X') - \frac{e^{\lambda_i}}{e^{\lambda_i} + 1} \right] \\
&= \lambda_i S_i - T \frac{e^{\lambda_i}}{e^{\lambda_i} + 1} \\
&= 0
\end{aligned} \tag{2.5}$$

Here, S_i is the number of substitutions i found in the data set, and T is the total number of pair of compounds (X, X') occurring in the same prototype in the data set. For those λ_i which correspond to unobserved substitutions, λ_i is arbitrarily set to the low value of -10.

We can extract the probability $P(i_2|i_1)$ of the ion i_2 substituting into a particular site in a crystal structure, given that ion i_1 is known to exist on that site. Letting X_n denote the n^{th} site of compound X,

$$\begin{aligned}
P(i_2|i_1) &= \frac{p(X_n = i_1, X'_n = i_2)}{p(X_n = i_1)} \\
&= \frac{e^{\lambda_{i_1,i_2}}}{1 + e^{\lambda_{i_1,i_2}}} \frac{1}{\sum_j \frac{e^{\lambda_{i_1,j}}}{1 + e^{\lambda_{i_1,j}}}}
\end{aligned} \tag{2.6}$$

Finally, we define the ionic substitution similarity of two ions as:

$$Sim_{ion}(i_1, i_2) = \begin{cases} \max(P(i_2|i_1), P(i_1|i_2)), & \text{if } i_1 \neq i_2 \\ 1, & \text{else} \end{cases} \tag{2.7}$$

The ionic substitution similarity function is defined to be the maximum of the two conditional probabilities $P(i_2|i_1)$ and $P(i_1|i_2)$ to allow for sampling deficiencies in the

data set. For example, consider the scenario in which ion i_1 is rare and appears in only 1% of the data set, ion i_2 is common and appears in 50% of the data set, and for every compound in which ion i_1 appears, another compound appears in the same prototype and composition, only with ion i_2 substituted for ion i_1. Choosing the minimum of the two conditional probabilities penalizes the probability of substitution because ion i_1 is rare. In this implementation, in which we believe our data set features a large number of underrepresented ions, we chose to err on the side of generosity and use the maximum possible ionic substitution similarity.

We gratefully acknowledge the contributions of Stephen Dacek and Shyue-Ping Ong, whose elegant code prototyped the data sets used in this thesis.

2.4 Fitting Parameters for Usage

Executing the above algorithm on an oxide data set (described in Chapter 3) results in the ionic substitution similarity function shown in Figure 2.2. The resulting function is heavily weighted towards zero, with the vast majority of our values occurring below 0.30.

Taking a logarithm of every value and re-scaling our similarity linearly such that all values lie within the interval [1, 0], we produce the logarithmic substitutional similarity shown in Figure 2.4. This similarity function is given by Equation 2.8. Again, the logarithmic similarity function satisfies $0 \leq \mathrm{Sim}_{\log \mathrm{ion}}(i_1, i_2) \leq 1$, and $\mathrm{Sim}_{\log \mathrm{ion}}(i_1, i_2)$ grows monotonically with the probability that i_1 and i_2 substitute for each other within the same prototype. However, the $\mathrm{Sim}_{\log \mathrm{ion}}$ better utilizes the full range of similarities from 0 to 1.

$$\mathrm{Sim}_{\log \mathrm{ion}}(i_1, i_2) = -\frac{\log \mathrm{Sim}_{\mathrm{ion}}(i_1, i_2) - \min\limits_{j_1, j_2} \log \mathrm{Sim}_{\mathrm{ion}}(j_1, j_2)}{\min\limits_{j_1, j_2} \log \mathrm{Sim}_{\mathrm{ion}}(j_1, j_2)} \tag{2.8}$$

The ionic substitution similarity function shown in Figure 2.6 finds two areas of high substitution rates consistent with intuition. The rare earths are clearly distinguished in the bright yellow lower left-hand corner, and the transition metals, roughly in the center of the diagram, also substitute with each other with high probability.

Both the original similarity function and the logarithmic similarity functions were tested in the development of the composition similarity function described in Chapter

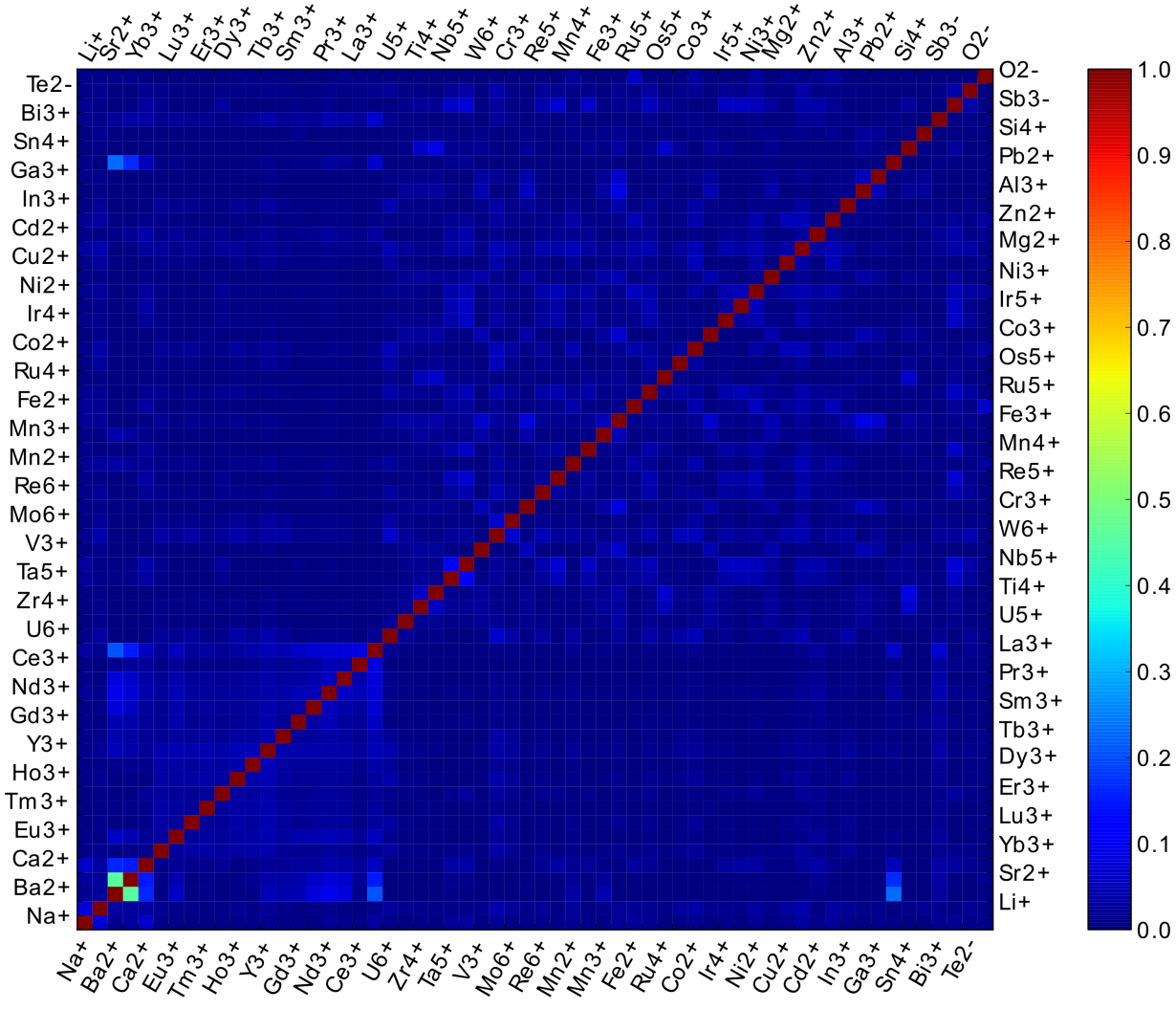


Figure 2.5: The probability of ion i_1 substituting for ion i_2 within the same structure prototype as given by the maximum likelihood model applied to the oxides in the ICSD. The sixty most common species in the data set are shown, ordered by Mendeleev number.

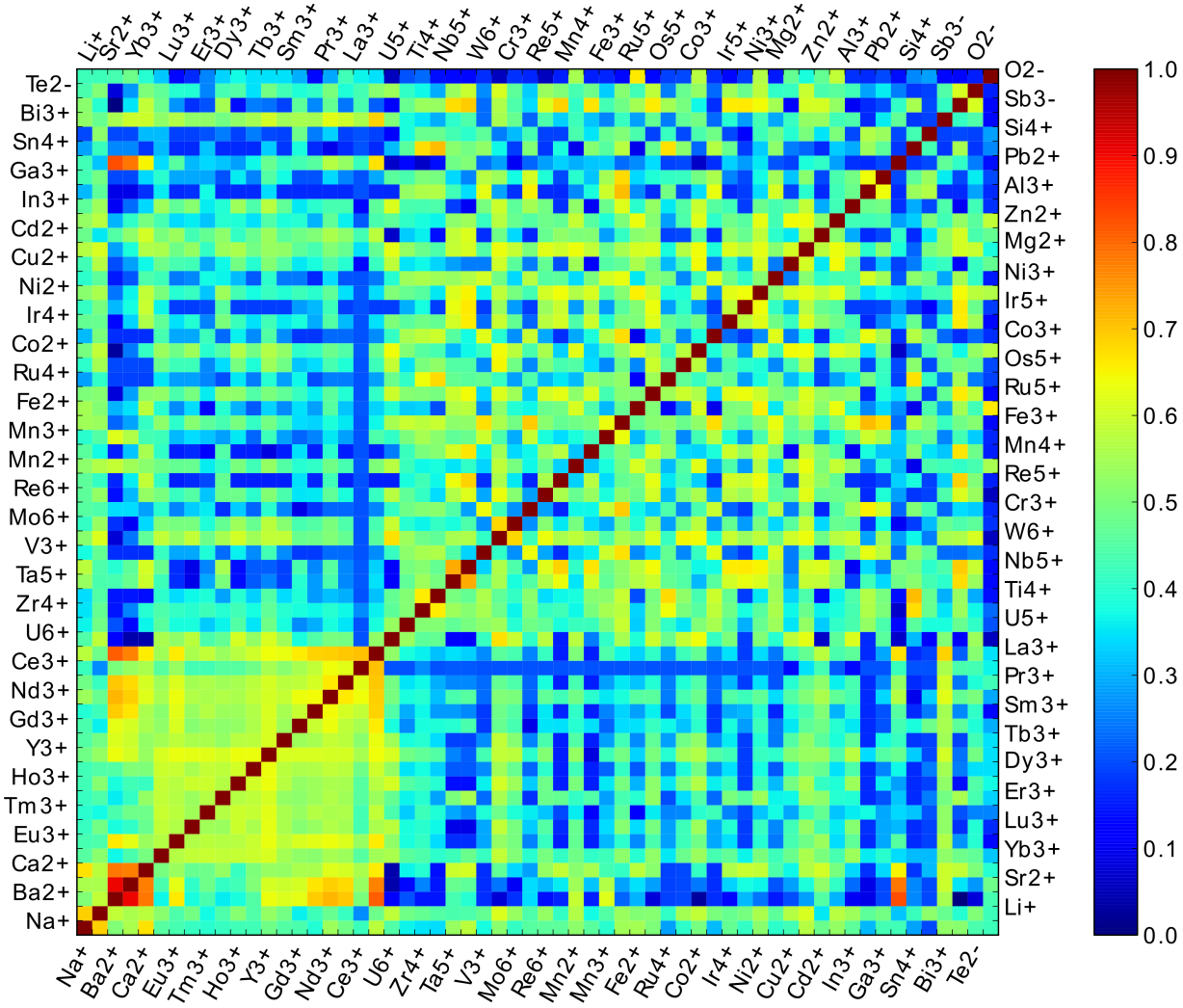


Figure 2.6: The probability of ion i_1 substituting for ion i_2 within the same prototype, re-scaled to better differentiate substitution probabilities. The sixty most common species in the data set are shown, ordered by Mendeleev number.

3, and the logarithmic similarity function was used to generate the superior results shown throughout Chapter 3.

Finally, it is occasionally useful to further tune the sensitivity of the ionic similarity function. In Chapters 4 and 5, we weigh ionic substitution similarity against geometric similarity. In these cases, we tune the sensitivity of the logarithmic ionic substitution similarity function by passing it through a sigmoid function; this sigmoid function takes the form of an integrated Gaussian:

$$\text{Sigmoid}_{\sigma,\mu}(x) = \frac{1}{\sigma\sqrt{2\pi\sigma}} \int e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{2.9}$$

Setting μ sets the threshold above which ions are considered similar, and setting σ sets how sharply the function differentiates similar and dissimilar ions.

The tunable ionic substitutional similarity function used throughout Chapters 4 and 5 is given by:

$$\text{Sim}_{\text{ion}}^{\sigma,\mu}(i_1, i_2) = \text{Sigmoid}_{\sigma,\mu}(\text{Sim}_{\log \text{ion}}(i_1, i_2)) \tag{2.10}$$

2.5 Clustering

The construction of a similarity function leads naturally to a few questions. With respect to the Sim_{ion} function, we may ask which ions are the most similar, and how similar are they? Can the ions be naturally grouped or *clustered* into families of similar ions? How many such families are there? These questions relate to the topology imparted on the space of ions by the Sim_{ion} function. In this section, we discuss clustering, a well-established problem within the computer science community. We will outline the details of a hierarchical clustering algorithm, and as an example, we will cluster the ions in our database using the ionic substitutional similarity.

The vast majority of clustering algorithms are based upon maximizing some measure of similarity within a cluster, or minimizing the parallel concept of distance. Throughout this thesis, we develop a number of similarity functions on sets X, $s\colon X \times X \to \mathbb{R}$. These similarity functions s will all satisfy the following properties for all $x_1, x_2 \in X$:

- $0 \leq s(x_1, x_2) \leq 1$ (non-negativity)

- $s(x_1, x_2) = 1$ if and only if $x_1 = x_2$ (identity), and
- $s(x_1, x_2) = s(x_2, x_1)$ (symmetry).

It is natural to use the following definition of distance:

$$\text{distance} = 1 - \text{similarity} \tag{2.11}$$

It follows that our distance functions will satisfy the following criterion for all pairs $x_1, x_2 \in X$:

- $d(x, y) \geq 0$ (non-negativity)
- $d(x_1, x_2) = 0$ if and only if $x_1 = x_2$ (identity)
- $d(x_1, x_2) = d(x_2, x_1)$ (symmetry)

However, our distance functions do not satisfy the triangle inequality

- $d(x_1, x_3) \leq d(x_1, x_2) + d(x_2, x_3)$,

which notably reduces the number of available clustering algorithms, the majority of which require the distance function at hand to be a metric satisfying the triangle inequality.

Throughout this thesis, we will use hierarchical clustering to display similarity data. Hierarchical clustering builds a hierarchy of clusters; the largest cluster contains the entire set, whereas the smallest clusters consist of a single observation. Each smaller cluster is fully contained in a larger cluster. Hierarchical clusters are displayed in dendrograms, several examples of which are shown in Figure 2.7.

Hierarchical clustering algorithms can operate on semi-metrics, and require the choice of a linkage function that determines the distance between sets of observations. The selection of the linkage function can strongly affect the outcome. A few common linkage functions between clusters A and B include:

- $\max\{d(a, b) : a \in A, b \in B\}$, known as complete linkage clustering
- $\min\{d(a, b) : a \in A, b \in B\}$, known as single linkage clustering, and

- $\frac{1}{|A||B|}\sum_{a\in A}\sum_{b\in B} d(a,b)$, known as average linkage clustering.

For the purposes of this thesis, we use complete linkage clustering in our hierarchical clustering algorithms. Complete linkage clustering guarantees that in every cluster A, every pair of observations $a_1, a_2 \in A$ satisfy $d(a_1, a_2) \leq d_{max}$ for some maximal distance. This strongest linkage criterion most severely limits the distances within a cluster, yielding a more intuitive understanding of the space being clustered.

Hierarchical clustering algorithms fall into two categories, divise and agglomerative. Divisive clustering algorithms, or top-down algorithms, begin with one cluster containing the entire space. However, exhaustive divisive clustering algorithms tend to operate in $\mathcal{O}(2^n)$ time, making them computationally infeasible. Agglomerative clustering algorithms start from the bottom up, with each element starting its own cluster; clusters are iteratively merged together. Agglomerative clustering algorithms operate in $\mathcal{O}(n^3)$ time. For the relatively small data sets in this thesis, comprising at most 10,000 data points, this is acceptable.

We gratefully acknowledge the SciPy.cluster package[45] for our implementation of the hierarchical clustering algorithm.

Figure 2.7 depicts the results of hierarchical clustering by ionic substitution similarity for the sixty most common ions in the data set in dendrogram form. The bracket that connects each pair of ions forms at the distance d_{max} at which these ions are clustered together. Red brackets indicate clusters with large diameters d_{max}; blue brackets indicate clusters of small diameter. Clusters of small diameter contain similar ions.

The effect of re-scaling the ionic substitution similarity is clearly observed. While the observations in each cluster remain the same, the diameter of each cluster is shifted. $\mathrm{Sim}_{\mathrm{ion}}$ favors clusters with large diameter, assigning most pairs of ions a very low similarities, while $\mathrm{Sim}_{\log \mathrm{ion}}$ uses a wide range of similarities. $\mathrm{Sim}_{\mathrm{ion}}^{\sigma,\mu}$ breaks the ions into six distinct, highly dissimilar groups, while ions in each group have very high similarity.

2.6 Discussion and Extensions

The construction of the binary indicator model allows for the existence of charge-imbalanced substitutions, as illustrated in Example 2. The λ_i are calculated based upon a count of the number of times a species i_1 substitutes for another species i_2 within the

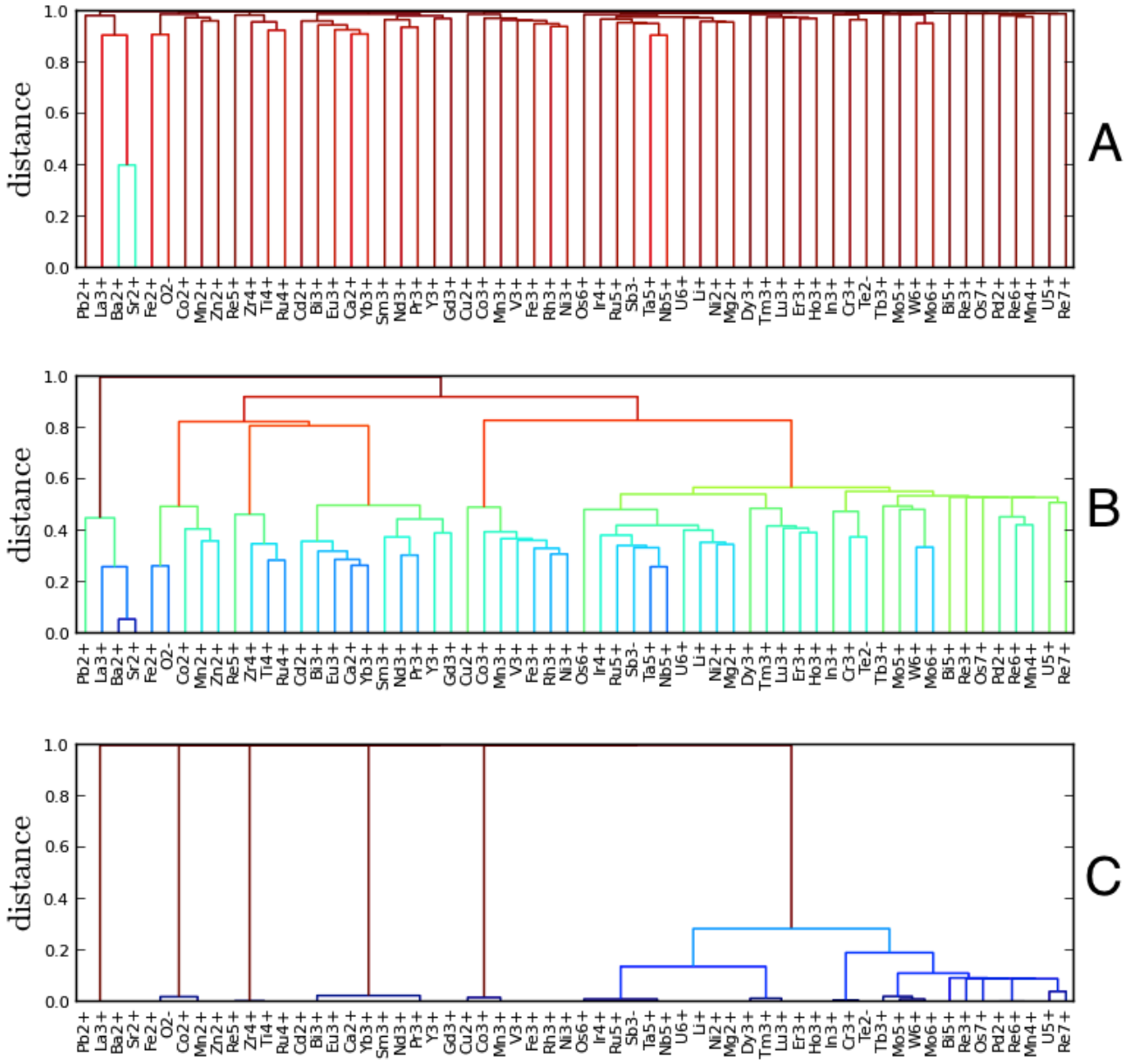


Figure 2.7: The hierarchical clustering of ionic species is shown for the sixty most common ions in the data set. $\mathrm{Sim}_{\mathrm{ion}}$ is shown in figure A, $\mathrm{Sim}_{\log \mathrm{ion}}$ in figure B, and of $\mathrm{Sim}_{\mathrm{ion}}^{\sigma,\mu}$ in figure C. The re-scaled ionic similarities do not change the observations in each cluster, but they do affect the diameters of the clusters.

same crystal structure prototype regardless of what other charge compensating substitutions may occur on other sites. Looking carefully through the transition metal substitutions found in the center of Figure 2.6, we find many examples of charge imbalanced substitutions occurring with high probability. This illustrates the capacity of this model to capture subtleties in the similarity of ionic species beyond that of charge state or ionic radius. With respect to the prediction of crystal structure, the construction of this ionic similarity function cuts straight to the heart of the problem; similarity is defined with respect to how often two ions will inhabit the same structure in the same site.

The existence of charge-imbalanced substitutions, which require a concurrent charge compensating substitution, clearly violates the assumption that the λ_i are independent. The model described by Hautier et al[36] can be thought of as a first order model; it only allows for binary interactions. Like a cluster expansion, its accuracy can be improved by including second order interaction terms; the addition of such $\lambda_{i,j}$ terms would greatly increase the number of parameters to be fit, and would require more data to fit robustly.

Finally, one weakness of the algorithm constructed in this chapter is that it is unable to distinguish between substitutions that do not occur because they are energetically unfavorable and substitutions that do not occur because they have not been reported in the data set at hand. Elements that are rare, expensive, or toxic tend to occur less frequently in experimentally reported databases, and thus tend to have low ionic substitutional similarities. This weakness is due to the fact that this algorithm is trained on positive data only. One potentially interesting but computationally expensive extension of this work would be to use *ab initio* methods to perform a statistical sampling of ionic substitutions. It would then be possible to assign definitive penalties, based upon calculated energies, to the rarely reported substitutions.

3

Composition Similarity

Growing materials databases and the availability of more computational power have lead to considerable development in computational materials design.[60,41] Growing databases of materials knowledge incur the necessity to develop methods with which to organize such knowledge. For instance, the Inorganic Crystal Structure Database now contains 161,030 entries.[7] Given a promising compound with certain properties, how can we systematically search for similar compounds? Such a definition of similarity must be with respect to a given property; in this chapter we develop a similarity function between two compositions that reflects the likelihood with which two compounds will have the same crystal structure.

Traditionally, materials scientists have relied upon structure mapping methods combined with heuristic design rules derived from the physical properties of individual ions to predict the structure of novel materials. For example, the Hume-Rothery rules relate the ratio of valence electrons per atom to the crystal structures formed,[39] while the

Pauling rules relate the structure of ionic materials to the radii of the ions involved,[75] and Pettifor maps demonstrate clustering of similar crystal structures in a space where the coordinate of each element is simply related to its position in the periodic table.[78] Miedema related the electron concentration at the boundary of a Wigner-Seitz cell to the formation enthalpy of binary metallic systems.[62] While these methods provide some physical insight, their predictive quality is limited,[67] and no obvious extension of these methods exists to make them more accurate or extend them to higher component systems.

Modern structure prediction methods may involve an unbiased search through the vast space of possible atomic arrangements;[28,47,94,95] others use chemical knowledge, including geometric data such as expected bond lengths,[49] secondary building units,[59] and structure prototype databases in conjunction with thermodynamic data[24] to reduce the search space. The problem with modern structure prediction methods is that they require a large number of energy evaluations, making them computationally costly.

Motivated by D.G. Pettifor's structure maps which displayed a correlation between ions with similar Mendeleev number and the binary structure prototypes in which they formed, we generalize Pettifor's idea to incorporate information from not only binary, but ternary, quaternary, and more complex compounds. Following the work of Hautier et al,[37,36] we use the ionic substitutional similarity function derived in Chapter 2 to develop a composition similarity function. This composition similarity function has the advantage over traditional, heuristic methods that it is general: any two compositions, regardless of the number or identity of the components, can be compared to each other. Indeed, knowledge gleaned from the binary and ternary systems is used to inform our knowledge of the quaternary and more complex systems. This generality can be used to impart a distance-like structure to the database of knowledge, clustering together compounds of similar composition. The composition similarity function has the advantage of speed over that of modern structure prediction methods; informed by knowledge gleaned from current structure databases, composition similarity correctly classifies a new composition by selecting the correct prototype for an oxide structure from a list of known prototypes within 5 guesses 80% of the time.

The search through the space of possible atomic arrangements becomes much more difficult when considering complex materials such as the quaternary oxides. This vast,

sparsely sampled search space, with its enormous number of combinatoric possibilities, represents a rich area for the development of new materials. We present in this chapter composition similarity, a data mined function on the composition of a material that uses chemical knowledge from binary and ternary compounds. We apply composition similarity to the problem of structure prediction, validating the hypothesis that composition similarity reflects structural similarity, and achieve remarkably successful results with respect to the complex oxides.

3.1 Similarity between Compositions

In this section, we develop a similarity function between two compositions. Note that "similarity" refers to a property for which these compositions behave similarly. In this case, that property is crystal structure. We begin by data mining from the oxide training set an ionic substitution similarity function between two ions, per Hautier et al[36], described in Chapter 2. We use the ionic substitution similarity as an input to the composition similarity function, defining a composition similarity function via the best matching of ions in composition c_1 to ions in composition c_2. The desired function should increase monotonically from 0 to 1 with the probability that the two compositions take the same prototype. It should achieve its maximum value of 1 when the two compositions are identical.

3.1.1 Maximum Matchings

Given the ionic substitutional similarity defined in Chapter 2, it is possible to define a quantitative data-mined similarity rating between two compositions. Calculating composition similarity involves finding the best possible matching of the ions in one composition to the ions in the other such that the average ionic substitutional similarity between each pair is maximized. As the concept of a best matching is used multiple times in this thesis, we include here a brief primer on maximum matchings.

A *graph* is an ordered pair $G = (V, E)$ comprising a set of V of vertices together with a set E of edges. Each edge $e = (v_1, v_2)$ represents a relationship between two vertices.

A *weighted graph* has a weight w_e assigned to each edge: $E : V \times V \to \mathbb{R}$.

Figure 3.1: A weighted bipartite graph. The orange nodes represent the nodes in U; the green nodes represent nodes in V. Edges must join nodes from U to nodes in V; no edge may connect two nodes of the same color.

A *bipartite graph* is a graph whose vertices can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V.

An example of a weighted, bipartite graph is shown in Figure 3.1. Bipartite graphs are often used when comparing objects that fall into two classes.

Within the context of this thesis, we will use bipartite graphs to represent relationships between species, specific ions within substructures, and between substructures. The nodes within our graphs typically represent ions; the edges represent similarities between the nodes. All the similarities in this thesis, and thus all the edges, will satisfy $0 \leq w_e \leq 1$.

A *matching* on a graph is a set of edges such that no two edges share a common vertex.

The maximum weighted bipartite graph matching problem, also known as the assignment problem,[18] is the problem of finding a matching on a bipartite graph such that the sum of the edge weights in the matching have the maximal (or minimal) value. One commonly used analogy when describing the assignment problem is that of workers and tasks. Given n workers, n tasks and the $n \times n$ set of non-negative costs for each worker to do each task, assign each worker a task such that each task is completed by a different worker with the minimal cost.

The assignment problem is commonly solved using the Hungarian algorithm[18] which operates in $\mathcal{O}(n^3)$ time. The Hungarian algorithm requires that the values in

the cost matrix are non-negative.

We gratefully acknowledge Brian Clapper for his open source Python implementation of the Hungarian algorithm which we have used extensively throughout this thesis.[16]

3.1.2 Composition Similarity Definition

A composition is defined as a set of ions $\{i\}$ together with the number of times each ion appears $\{n\}$. We represent compositions by their reduced versions, where the reduced version has the smallest integers $\{n\}$ that preserve the correct ratios between the ions. The sum Σn of the number of ions in the reduced composition is the total number of ions n_{total} of a given composition.

Given two compositions c_1 and c_2, we find the lowest common multiple n_{lcm} of n^1_{total} and n^2_{total}. Two sets of ions s_1 and s_2 of length n_{lcm} are created by enumerating the ions of c_1 and c_2 the appropriate number of times. Searching through all the possible matchings (i_1, i_2) of the ions i_1 in s_1 to the ions i_2 in s_2, the matching that maximizes the average similarity of the two sets is found. This maximal average similarity is defined as the composition similarity between c_1 and c_2.

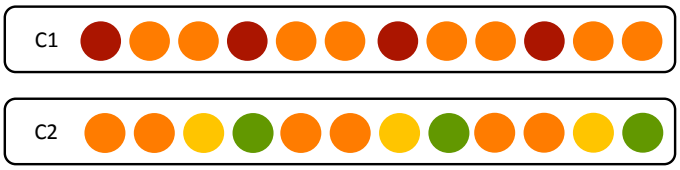
$$\mathrm{Sim}_{comp}(c_1, c_2) = \max_{\text{all matchings}} \frac{\Sigma_{i_1, i_2 \in \text{matching}} \mathrm{Sim}_{ion}(i_1, i_2)}{n_{lcm}} \tag{3.1}$$

For clarity, we depict in Figure 3.2 a schematic of a sample calculation.

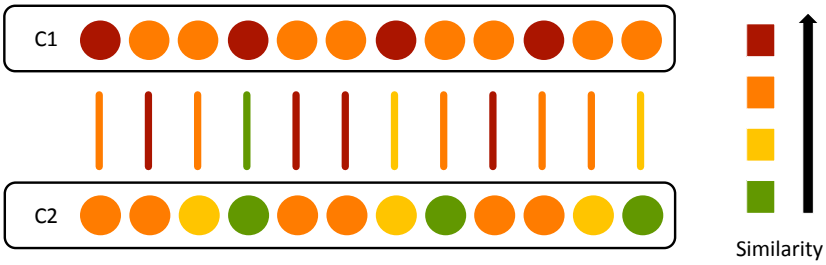
To limit computational complexity, for the purposes of this thesis we cap n_{lcm} at a maximum value of 100. The implementation of this cap has several subleties which we detail here. If $n_{lcm} > 100$, we must ascertain how many ions from each composition to compare to each other. This number serves as the divisor when we calculate the maximal average ionic similarity. We wish to guarantee the representation of at least one full reduced composition for each composition c_1, c_2. Therefore, if either n^1_{total} or n^2_{total} is greater than the cutoff value of 100, we will compare the larger number of ions $\max(n^1_{total}, n^2_{total})$ from each composition. Otherwise, the divisor will be a whole number of multiples of either n^1_{total} or n^2_{total} to each other, such that the divisor is as close as possible to 100, but still less than or equal to 100. Pseudo code is given for generating the divisor below.

Given two compounds with compositions C1 and C2,

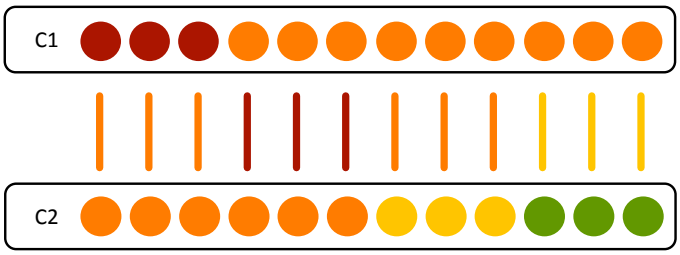
Step 1: Find the lowest common multiple of the number of ions in each:



Step 2: Assign each pair of ions a similarity:



Step 3: Rearrange the ions to maximize the average similarity:



which we define to be the composition similarity between C1 and C2

Figure 3.2: Composition similarity algorithm. The composition similarity is calculated by finding the maximum matching of ions in one composition to the ions in another. The weights of the edges are given by the ionic substitution similarities between the ions. The maximum matching is calculated between the same number of ions from both compositions.

```
currentLCM = lcm(numSpecies1, numSpecies2)
if (numSpecies1 > cutoff) or (numSpecies2 > cutoff):
 divisor = max(numSpecies1, numSpecies2)
else if (currentLCM <= cutoff):
 divisor = currentLCM
else:
 if (cutoff% numSpecies1 > cutoff % numSpecies2):
 multiplier = int(cutoff)/int(numSpecies2)
 divisor = multiplier * numSpecies2
 else:
 multiplier = int(cutoff)/int(numSpecies1)
 divisor = multiplier * numSpecies1
```

The composition similarity yields a rating between 0 and 1 for every pair of compositions c_1 and c_2, with identical compositions having similarity 1. Based on data mined values for the probability with which each ion will substitute for another within the same prototype, this composition similarity provides a quantitative method to evaluate the likelihood with which two compounds will form in the same prototype.

3.2 Application to the Prediction of Structure Prototypes

In this section we validate the composition similarity function by assessing the extent to which compounds with similar compositions are likely to appear in the same prototype. The structure prediction problem depends upon the assumption that at a given set of external conditions, the composition of a compound determines its structure. Therefore we must assess whether or not the relationship between composition and structure is captured by the proposed composition similarity function.

We begin by describing the construction of the data set used in this chapter. We will examine in detail the effect of clustering by composition similarity on three example compounds from the data set. We then present a statistical analysis of the overall performance of prototype prediction by composition similarity.

3.2.1 Data Set Construction

The data set that we used to validate composition similarity was constructed with several criterion in mind. The data set should be of moderate size; large enough to data mine patterns from, but small enough to be computationally feasible on a trial basis. It should contain a variety of prototypes and chemistries, but for an initial validation, this variety should be contained within a relatively dense space of prototypes. Finally, the data set should be of interest to the scientific community.

We selected the family of oxide crystal structures for use in this thesis, as it comprises a well-studied, highly structured, and commercially pertinent family of compounds. Furthermore, we removed peroxides, superoxides, high temperature and high pressure phases in an effort to limit our data set to that of the ionic, low-temperature and low-pressure solids, thus limiting the complexity and the sparsity of the data set at hand. Lastly, we subjected the data set to a rigorous cleaning process to remove structures which could have been incorrectly reported; we removed structures with hydrogen, structures with reported composition-structure mismatches, unbelievably short bond lengths, and duplicate structures.

All of the compounds in the Inorganic Crystal Structure Database[7] (ICSD) 2012 were searched for compounds that satisfied the following criteria:

- Compounds must be oxides, as indicated by at least 20% oxygen content by ion count.
- Compounds must not be peroxides or superoxides, as indicated by O-O bond lengths L < 1.50 Å.
- Compounds must not be marked 'high pressure', 'HP', 'high temperature', or 'HT.'
- Compounds must not have improbably short (< 1Å) bond lengths.
- Compounds must not have a mismatch between the reported composition and the ions given in the crystal structure.
- Compounds must not contain hydrogen. The reported crystal structures of compounds containing hydrogen are often unreliable.

The resultant oxides were sorted into structure prototypes using an affine mapping algorithm.[40,11] The data set was further cleaned by removing duplicates, defined as compounds with the same composition and the same structure prototype, resulting in a final data set of 5,694 oxide compounds.

3.2.2 A Few Examples

After cleaning, the complete data set was randomly split into two sets for cross-validation. The first set, called the training set, consists of 95% of the compounds, representing the database of known compounds to be data mined. The remaining 5%, called the test set, serves as a set of as-yet-unseen compounds which we used to evaluate the efficacy of composition similarity. We performed the cross validation process a total of 5 times.

The ionic substitution similarity function was evaluated based upon the compounds in the training set. Using the ionic substitution similarity, composition similarities were calculated between every composition in the test set and every compound in the training set. Finally, for each composition in the test set, the compounds in the training set were ordered by composition similarity. This cross-validation process was completed with five different partitions of training and test sets; here, we present the cumulative results from all 5 partitions.

We begin by examining the behavior of clustering by composition similarity upon three sample compounds from the test set.

Example 1: Ca_2FeWO_6

Table 3.1 and figure 3.3 summarize the results of sorting by composition similarity to Ca_2FeWO_6, which appears in the test set in the double perovskite crystal prototype. Ranking the compounds in the training set by composition similarity to Ca_2FeWO_6, we find that composition similarity begins by finding compounds that differ by one ionic substitution per formula unit. The first most similar compound substitutes Mo for W, yielding Ca_2FeMoO_6, which forms in a distorted double perovskite prototype. The next two most similar compounds are two polymorphs, Ca_2NiWO_6, forming in the distorted double perovskite prototype and an experimentally reported prototype featuring square planar-coordinated Ni^{2+} ions. The next two most similar compounds, Ca_2MnWO_6, Ca_2MgWO_6, both form in the distorted double perovskite

Table 3.1: Prototype Prediction for Ca_2FeWO_6

Composition	Prototype	Similarity to Ca_2FeWO_6
Ca_2FeMoO_6	1	0.968
Ca_2NiWO_6	1	0.965
Ca_2NiWO_6	2	0.965
Ca_2MnWO_6	1	0.961
Ca_2MgWO_6	1	0.961
Ba_2FeWO_6	1	0.956
Ba_2FeWO_6	3	0.956

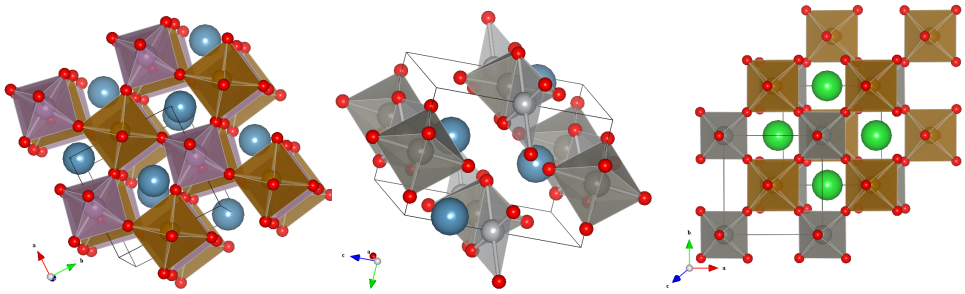


Figure 3.3: Three structure prototypes suggested for Ca_2FeWO_6. The first and third prototypes are both double perovskites, with the octahedron in the first prototype being slightly distorted. The second prototype represents an experimentally determined polymorph of Ca_2NiWO_6.

prototype. Finally, the next guess, Ba_2FeWO_6, representing a substitution of two Ba^{2+} ions for Ca^{2+} ions per formula unit, appears in two polymorphs; the distorted and perfect double perovskite prototypes. In this example, the composition similarity method finds the correct structure prototype for the compound in question on the 3rd guess., though all guesses are structurally similar to Ca_2FeWO_6.

Example 2: $BaLa_2Ti_3O_{10}$

Table 3.2 summarizes the results of sorting by composition similarity to $BaLa_2Ti_3O_{10}$, which appears in the test set. The most similar compound in the training set is $La_4Ti_3O_{12}$, followed by $La_2Ti_2O_7$. With the third guess, composition similarity returns to the original stoichiometry and finds $BaPr_2Ti_3O_{10}$, which forms in the same prototype as $BaLa_2Ti_3O_{10}$.

Table 3.2: Prototype Prediction for $BaLa_2Ti_3O_{10}$

Composition	Similarity to $BaLa_2Ti_3O_{10}$
$La_4Ti_3O_{12}$	0.973
$La_2Ti_2O_7$	0.965
$BaPr_2Ti_3O_{10}$	0.962

Table 3.3: Prototype Prediction for $DyMnO_3$

Composition	Prototype	Similarity to $DyMnO_3$
$DyMnO_3$	1	1.000
$DyMnO_3$	2	1.000
$DyMnO_3$	3	1.000
$YMnO_3$	3	0.927
$YMnO_3$	4	0.927

Example 3: $DyMnO_3$

Table 3.3 and figure 3.4 summarize the results of sorting by composition similarity to a polymorph of $DyMnO_3$ that appears in the test set. The $LnMnO_3$ structures, where Ln is a lanthanide, form primarily in two structural prototypes – a distorted orthorhombic perovskite and a hexagonal structure[97]. The target compound in our training set forms in the distorted orthorhombic perovskite structure, shown in figure 3.4.

Examining the results of ordering the test set by composition similarity shown in Table 3.3, we find that the three most similar compounds in the test set are polymorphs of $DyMnO_3$; two distorted versions of the distorted orthorhombic perovskite structure and the hexagonal structure. Lastly, composition similarity substitutes Y for Dy, guessing two polymorphs of $YMnO_3$; the first polymorph is the tetrahedral arrangement, and the second is the correct distorted orthorhombic perovskite.

Again, composition similarity selects crystal structures that have similar structural components. Three out of the first four structures are remarkably similar distorted orthorhombic perovskite structures; the last is a well-known polymorph of $DyMnO_3$. The final structure only differs from previous ones by a slight shift in the placement of the central Dy^{3+} ion.

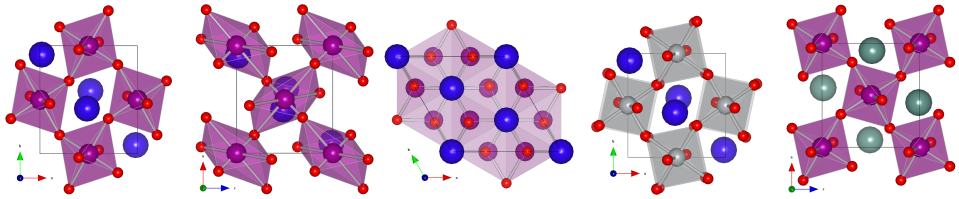


Figure 3.4: Four structure prototypes suggested for $DyMnO_3$. Numbers 1, 2, and 4 are all distorted orthorhombic perovskite structures. Structure prototype number 3 is a hexagonal crystal structure commonly found in the $LnMnO_3$ family, where Ln is a lanthanide.

3.2.3 Statistical Results

To quantitatively assess the ability of composition similarity to cluster compounds with similar structure, we consider the application of composition to structure prototyping. For each composition in our test set, we ask - what structure prototype would this compound form in? We answer this question by referring to the list of compounds in our training set, ordered by similarity to the test set composition. Structure prototypes are guessed from that list, progressing from the most similar compounds downwards, subject to the following rules:

1. If a compound in the test set forms in a structure prototype that is unrepresented in the training set, we do not consider this compound, as it is impossible to guess this structure prototype. Composition similarity does not have the ability to suggest as-yet-unseen crystal structures.

2. If the test set composition is a binary, ternary, or quaternary or more complex composition, only appropriate binary, ternary, or quaternary or more complex prototype guesses are permitted, matching the number of chemical components in the compound.

3. No prototype is guessed twice.

4. No training set compounds with the same composition as the test set composition are considered. Recall that the entire dataset, consisting of both test set and training set, is constructed such that no two entries share the same composition and prototype; such entries would be considered duplicates. Thus, guessing

compounds with the same composition would be by definition suggesting polymorphs with incorrect prototypes.

In the following analysis our structure prototype prediction algorithm, the 'composition similarity' algorithm, is compared against a control in which the list of suggested prototypes is ordered by the frequency with which these prototypes appear in the training set, called the 'most common prototypes' algorithm. Prototypes are guessed, subject to the same rules, from the most frequently observed prototypes to the least. This is a similar benchmark to what was used previously by Fischer et al[24].

Figure 3.5 depicts the performance of crystal structure prototyping via composition similarity aggregated across all 5 cross-validated test sets. The horizontal axis depicts the probability with which the correct prototype is among our guesses against the vertical axis, which depicts the number of guesses made. The black line shows the performance of prototyping via composition similarity. The dotted line shows the performance of prototyping via the most common prototypes method. Prototyping via composition similarity consistently outperforms prototyping via the most common prototypes method, requiring a smaller number of guesses at every confidence level. Overall, prototyping via composition similarity achieves 80% accuracy within 5 guesses. 67% of the time, the correct prototype is guessed within the first 3 guesses.

The strengths and weaknesses of composition similarity become evident when the data is broken down by the number of components in the compound. Dividing the data set into groups of binary, ternary, and quaternary or higher compositions, a clear trend in favor of the prediction of more complex compound prototypes emerges. Figure 3.6 shows the number of guesses necessary to find the correct prototype, broken down by number of components in the compound, for prototypes ordered via the composition similarity and most common prototypes methods. The composition similarity rating fares poorly in the binary compounds, performing comparably to the most common prototypes method. Its performance improves markedly in the ternaries, while the most common prototypes method suffers from the large number (~1100) of candidate ternary prototypes. The trend continues into the quaternaries, where the most common prototypes ranking fares remarkably well, predicting the correct structure prototype out of over 1,600 candidate structure prototypes on the first guess 65% of the time, within 2 guesses 80% of the time, and within 10 guesses 90% of the time.

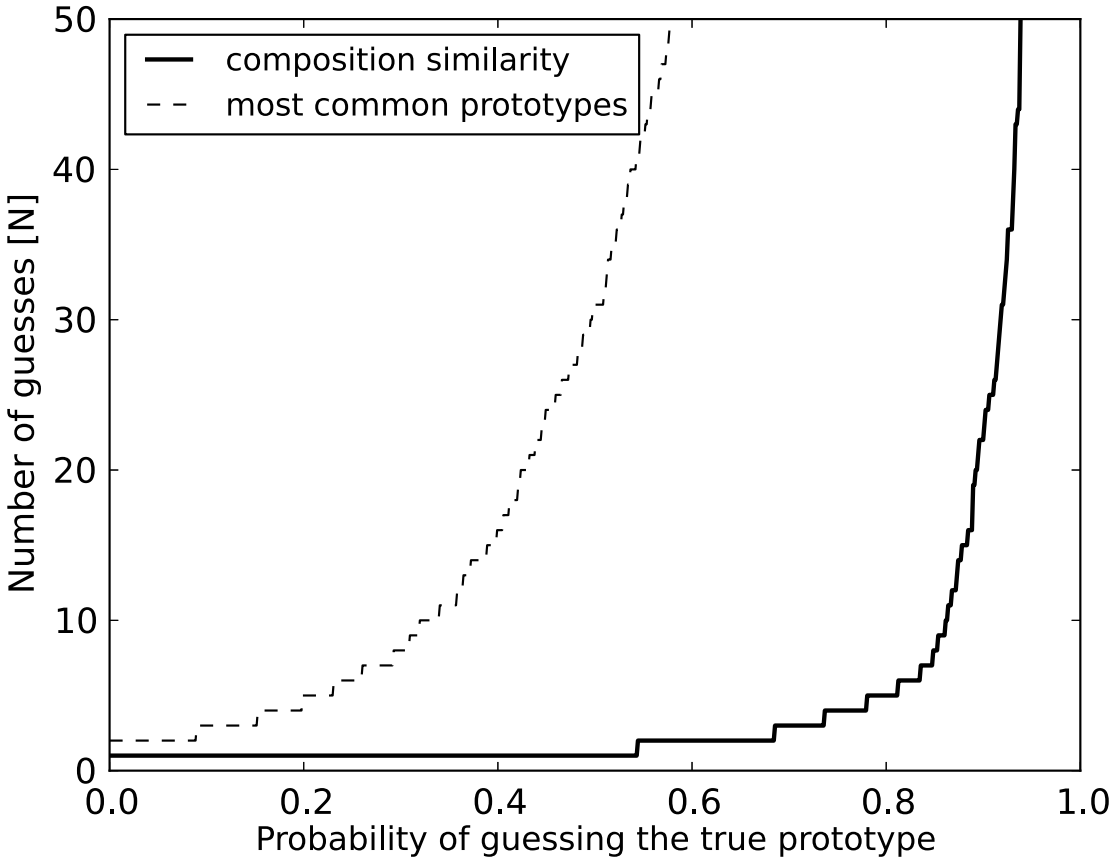


Figure 3.5: Prototyping ICSD oxides by composition similarity. The black line shows the number of guesses necessary to select the correct prototype by composition similarity. The dotted line shows the number of guesses necessary if those guesses were ordered by the frequency with which that prototype appears.

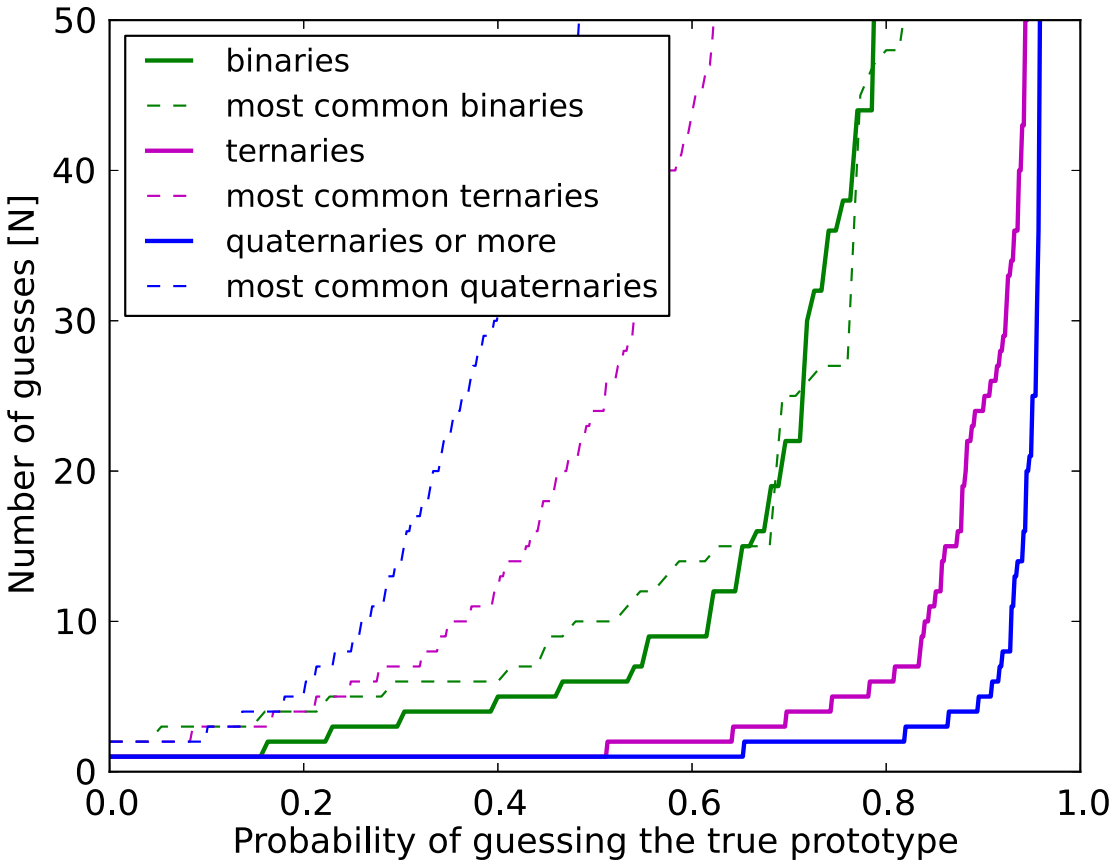


Figure 3.6: Prototyping oxides by complexity of compound. The bold lines show the number of guesses necessary to select the correct prototype by composition similarity. The dotted lines shows the number of guesses necessary if those guesses were ordered by the frequency with which that prototype appears. The performance of composition similarity increases dramatically with the number of components in a compound.

3.3 Discussion and Extensions

We have presented a data-mined, quantitative composition similarity function that reflects the probability of two compositions taking the same crystal structure prototype. This composition similarity function is obtained in two steps; the first is to data mine an ionic substitutional similarity function that reflects the probability that two ions will substitute for each other within the same prototype. The second is to use this ionic substitutional similarity to find the most similar matching of the ions in two given compositions; the average similarity of this matching is the composition similarity.

We have used structure prototype prediction as a means of evaluating the efficiency with which composition similarity groups similar compounds. Using composition similarity, we ordered the oxides in a training set of over 5,000 compounds versus each compound in a test set of over 280 compounds - five times. The compounds that appear first on the ordered list represent the most similar compounds to the test set compound in question. We have shown that these most similar compounds are very likely to share the same prototype as the test set compound, validating the hypothesis that similarity in composition correlates with similarity in structure.

It is worth noting that while structure prediction by composition similarity is dependent upon having a large, trustworthy data set that is pertinent to the structure being predicted, it also has the remarkable ability to predict crystal structures on a large scale. Compared to first-principles calculation based structure prediction algorithms, data mining is lighting fast, yielding the ability to predict large numbers of crystal structures. To our knowledge, no *ab initio* based structure prediction algorithms have published statistical evaluations of the efficacy of their methods.

We have found that composition similarity orders the possible prototype structures more effectively as the number of components of the compound increases, finding the correct prototype in remarkably few guesses. This is particularly useful for multiple reasons. Complex oxides are a particularly rich family of compounds comprising a large body of current scientific research. Additionally, complex compounds tend to have more atoms per unit cell; this hurdle is particularly difficult for first-principles structure prediction methods, which scale poorly with the number of atoms per unit cell. First principles methods are further limited by the necessity of guessing how many atoms there are per unit cell prior to calculation, whereas structure prediction by composition

similarity neatly circumvents this problem.

The correlation between performance of composition similarity and the number of components of the compound in question can be attributed to the relative lack of prototypes in the quaternary structures. Figure 3.7 shows the prototype distributions throughout an oxide data set. The x axis depicts the number of prototypes necessary to capture y percent of the compounds in the data set. Lines with low slopes depict prototypes that account for a small percentage of the structures in a data set; high slope regions represent prototypes that are densely populated.

While the binary compounds contain only one non-oxygen ion and form in over 280 distinct prototypes, the ternaries contain two non-oxygen ions and form in 970 prototypes, and the quaternaries contain three non-oxygen ions and form in 1300 prototypes. Considering the number of possible ionic combinations, with over 200 species of ions represented in our database, the number of combinations grow 200-fold going from the binaries to the ternaries and 4000-fold from the binaries to the quaternaries. However, the number of prototypes the algorithm must order in this data set grows far more slowly to the benefit of the predictive ability of our algorithm. While there undoubtedly exist ternary and quaternary prototypes that are as yet unrepresented in the ICSD – 50% of the quaternary prototypes in the test set were not represented in the training set - this study shows that composition similarity takes advantage of the higher ratio of ionic combinations to number of structural prototypes available in a quaternary composition to better order the current list of available prototypes. For those quaternary prototypes that were represented in the training set, we find the correct prototype on the first guess 65% of the time.

We refer back to example 3.2.2, in which we rank training set compounds by their composition similarity to $BaLa_2Ti_3O_{10}$, to highlight one of the weaknesses of composition similarity. In this example, it takes composition similarity three guesses to obtain the correct structure prototype. The first two guesses, $La_4Ti_3O_{12}$ and $La_2Ti_2O_7$, exhibit starkly differing stoichiometries from the desired compound which would make them unlikely candidates for sharing the same structure prototype. However, because composition similarity is given by the highest average chemical similarity between pairs of ions from both compounds, high chemical similarity between a majority of the ions can outweigh a few improbable substitutions. Looking carefully at this example, it would seem improbable for the ions from $BaLa_2Ti_3O_{10}$ to map stoichiometrically

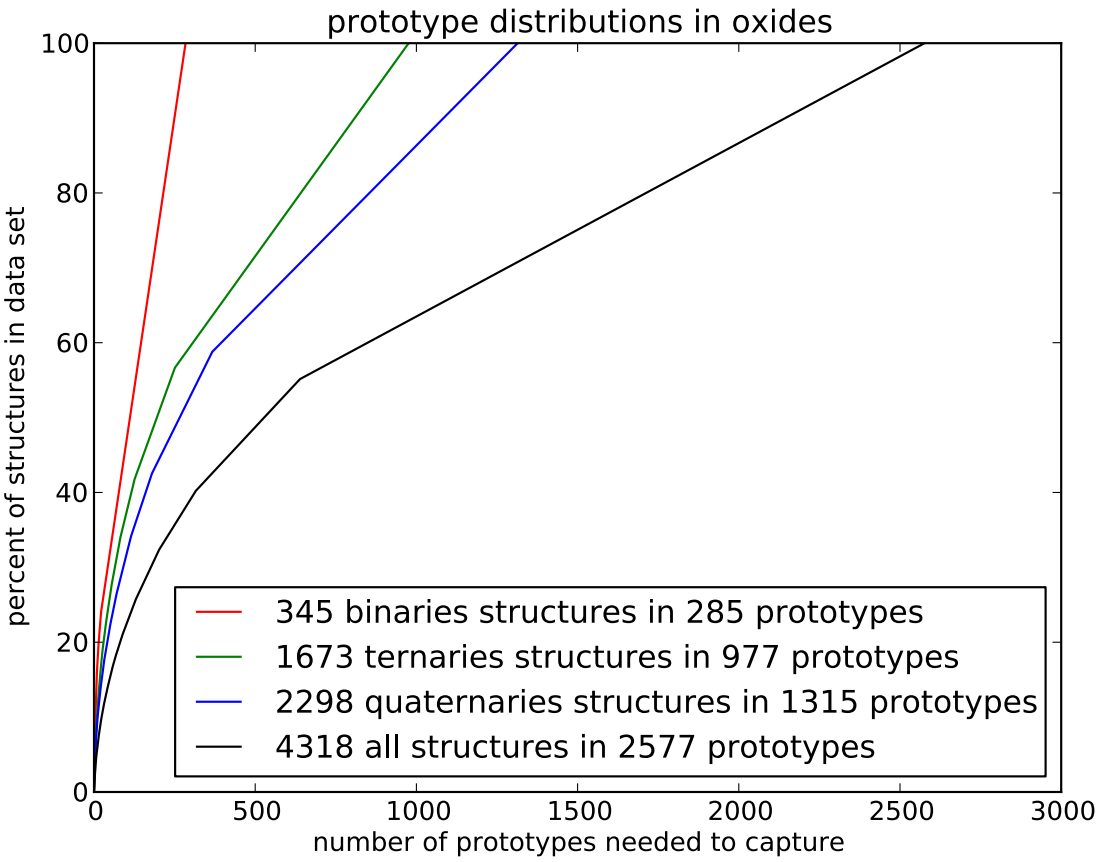


Figure 3.7: Prototype distributions in the oxides. 285 binary prototypes account for all binary crystal structures, whereas 1315 quaternary or more complex prototypes account for all the complex oxides. Approximately 400 quaternary prototypes appear more than once, accounting for almost 60% of the structures in the data set.

onto the ions of $La_4Ti_3O_{12}$; however, taking the lowest common multiple the 16 ions per formula unit in $BaLa_2Ti_3O_{10}$ and the 19 ions per formula unit in $La_4Ti_3O_{12}$, we find our similarity function is forced to compare 247 ions of each composition. In this case, one of the O^{2-} ions from $BaLa_2Ti_3O_{10}$ must eventually be mapped onto either La^{3+} or Ti^{4+}, a rather improbable substitution. However, over 247 comparisons in total, there are enough overwhelmingly good substitutions to outweigh a few improbable ones.

In future work it would be possible to address this weakness through a simple algorithmic variation. The substitution of ions in differing charge states could be strictly disallowed by automatically setting their similarities to $-\inf$. Such a modification, while computationally straightforward and physically meaningful, would also result in a loss of information. The current algorithm data mines chemical similarity while allowing for multiple substitutions within the same prototype; it is not uncommon for ions of differing charge states to substitute for each other when accompanied by another, simultaneous substitution which offsets the charge imbalance. The information gleaned from charge-imbalanced, multiple-ion substitutions would be lost if we implemented this variation.

Compared to other structure prediction algorithms, ordering via composition similarity has distinct strengths and weaknesses. Unlike direct optimization search methods, composition similarity does not have the ability to predict new structure prototypes that are not represented in current databases. Some direct optimization search methods, for example, simulated annealing, can attempt to systematically search through the infinitely-sized space of possible structures given infinite time, while composition similarity is strictly limited to searching the data set at hand. However, composition similarity effectively orders structure prototype candidates prior to any energetic evaluation, and is thus quite computationally cost-effective. Furthermore, the growing efficacy of composition similarity with the complexity of the compound makes structure prediction via composition similarity much more attractive in the quaternary or even quintenary compounds. We expect the performance of composition similarity to improve as more quaternary compounds are discovered.

Composition similarity has broader potential for application than the example of prototype prediction discussed above. The composition similarity function takes as its input any two compositions, and outputs a number that reflects the chemical and

structural similarity between them. Such a function is useful with respect to the classification: we demonstrated the performance of composition similarity when classifying compounds by structure prototype. However, in a broader sense, composition similarity is useful because it is an effective clustering method, grouping together compounds that are similar and providing a mechanism for the mapping of composition space. Defining the distance between two compositions as 1 – the composition similarity, we now have a semi-metric which imposes a structure upon the space of compositions. Note that this distance does not satisfy the triangle inequality, and thus is only a semi-metric.

Figure 3.8 shows a small sample set of 300 compounds drawn randomly from the test set, clustered by composition similarity. The vertical axis represents the distance between two compounds, or the largest possible distance between two clusters. Identical compounds (SiO_2) have distance zero. The horizontal axis represents the clustering of similar compounds; generally speaking, similar compounds will be drawn closer to each other on the horizontal axis.

Figure 3.8 represents the ability of composition similarity to provide structure to a well-explored but as yet relatively unmapped space; the space of all compounds. It can form a hierarchical grouping of similar compounds across all chemistries, comparing binary compounds to ternaries and quaternaries in a quantitative, physically meaningful way. We suggest that the organizational value of composition similarity may prove useful, allowing the designers of new compounds a new mechanism by which to search for compositionally similar compounds.

Finally, it is possible to extend the composition similarity method such that it is no longer based upon structural similarity. This chapter describes a two-step algorithm; the first part, following the work of Hautier[36], data mines an ionic substitutional similarity between two ions which reflects the tendency of those ions to form within the same structure prototypes. The second part describes how to use that ionic similarity function to compute a composition similarity function. Appropriately, we use the ionic similarity that reflects the tendency to form within the same prototype to predict crystal structure. However, the two parts are modular; the derivation of composition similarity is independent of which ionic similarity function is used. If the end goal were not the prediction of crystal structure but another property, using another ionic similarity function may prove more direct and yield a better prediction.

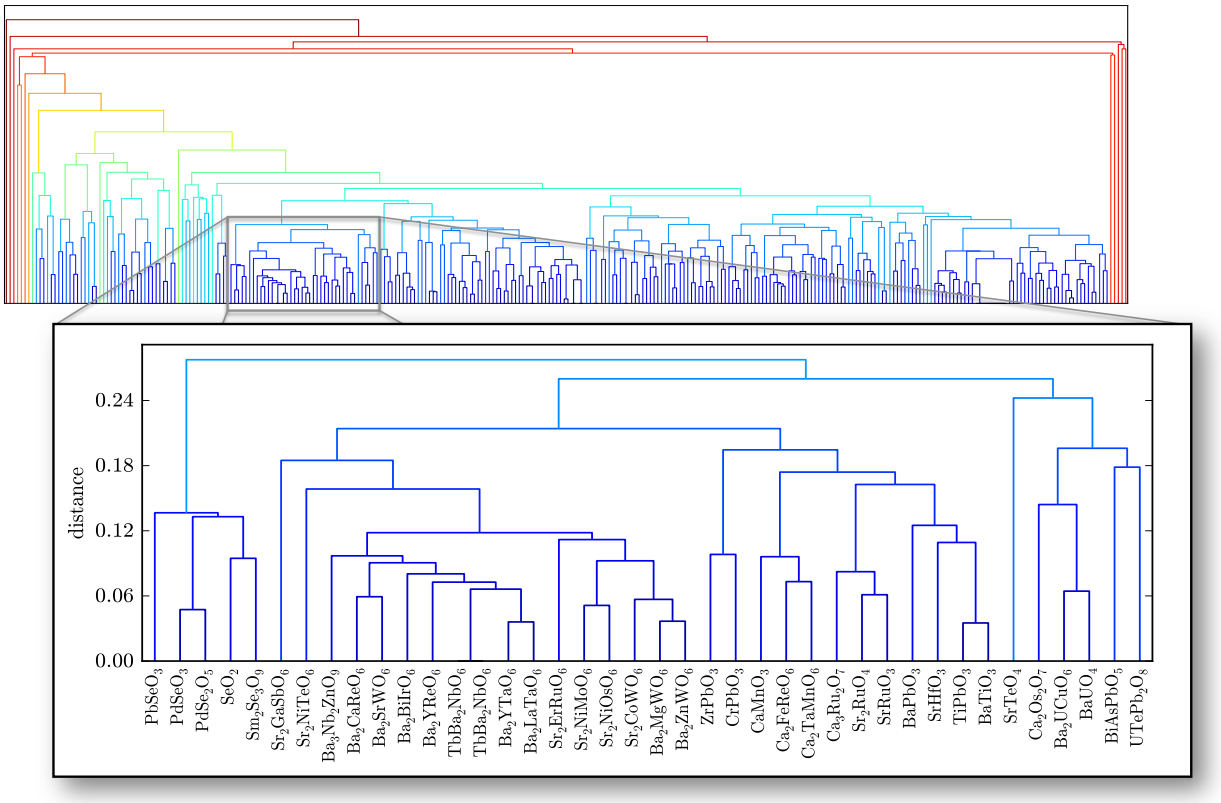


Figure 3.8: Clustering compounds by composition similarity. Grouping together compounds with high similarities imposes a structure on composition space.

Data mining across not only structural but also chemical similarity allows us to counteract the sparsely sampled nature of the space of quaternary materials; indeed, when limited to the quaternary subset of the oxides, composition similarity correctly selects the correct prototype from a list of known prototypes on the first guess 65% of the time.

4

Substructural Similarity

In the previous chapter, we developed a similarity between ionic compositions and used it to validate the correlation between similar compositions and crystal structure prototypes. Pursuant to the belief that ionic similarity is a rich tool for data mining structural materials properties, in this chapter we link ionic substitutional similarity to a structural property that is less rigidly defined than crystal structure prototype. Breaking crystal structures down into substructures allows us to further subdivide our data set, yielding more points in a richer database. Instead of mining 5,700 compounds in 2,600 prototypes, we are now able to extract millions of substructures from the same database of 5,700 compounds. Again, we use ionic substitutional similarity to construct a similarity function between substructures, this time including an explicit geometry-dependent term in our substructural similarity definition. This new similarity function allows us to combat data sparsity, collecting information from across binary, ternary, and quaternary prototypes to predict defect ion insertion sites.

Substructure based analyses of materials properties have a rich history in materials science. Linus Pauling's Principles determining the structure of complex ionic crystals[75], colloquially known as the Pauling rules, describe a set of rules for deriving ionic crystal structures based upon geometry and local packing rules. Pauling begins with a geometric argument based upon the ratio of the cation radius to the anion radius, fitting as many anions about a central cation as geometry will allow. Pauling continues on with various electrostatic arguments about the packing and arrangements of ionic polyhedra.

Daams and Villars presented in 2000[21] an enlightening study in which they decomposed 200,000 inorganic crystal structures in 5,000 structural prototypes into chemistry-independent atomic environment types. Promisingly, they showed that only 20 of the most frequent atomic environment types were necessary to account for 80% of the prototypes seen; only another 70 rarer atomic environment types were necessary to account for their entire data set.

Given the high degree of structure that Daams and Villars have found in the atomic environments of inorganic compounds, alongside Pauling's intuitive and compelling physical arguments, decomposing crystal structures into substructural units is a natural next step. Substructural decomposition is less rigidly constrained than crystal structure prototyping, allowing us to further subdivide the data set at hand. On the other hand, the relatively small number of geometrically distinct substructures we expect to find prevents us from exposing ourselves to intractably large numbers of substructures which would prove computationally expensive to run statistical analyses upon.

In this chapter, we validate our work with applications to the development of lithium ion batteries. Lithium ion batteries are commonly used in consumer electronics such as cell phones, power tools, and even electric cars. Lithium ions flow from anode to cathode and back during charge and discharge, intercalating into the electrode materials. The commercial success of lithium ion batteries has prompted a surge of computational battery research.[61] In particular, the Materials Project[42] presents an exhaustive toolkit for the exploration of inorganic compounds, providing amongst other data voltage profiles and oxygen evolution data for potential lithium ion battery electrode materials. However, one basic problem that the Materials Project has yet to develop a computationally feasible strategy to solve is this: In a potential lithium ion electrode material, where are the lithium ion sites?

It is possible to use first-principles *ab initio* methods to calculate the energy required to place an Li-ion on a specific site within a crystal structure. Indeed, this method is used to calculate Li-ion intercalation voltages.[15,3] However, such a computation requires the user to relax the crystal structure around the intercalated ion; such a 'single-site' calculation is not computationally cheap. Using first-principles methods to exhaustively search the space of potential Li-ion intercalation sites across all the inorganic crystal structures would be very computationally expensive, as many inorganic crystal structures have hundreds or even thousands of candidate Li-ion sites.

It is to such computationally difficult search problems that the data mining techniques in this thesis are most readily applied. Data mining techniques are probabilistic in nature and depend upon the completeness of the training set at hand; they cannot provide the relative reliability of an *ab initio* calculation. On the other hand, data mining techniques are incredibly fast. Thus it makes sense to preface any high-throughput series of *ab initio* computations with a data mining component which orders the search space by probability of success. In chapter 3, we ordered the search space of complex oxide prototypes to great success. In this chapter, we use data mining in conjunction with the substructural similarity function to order the search space of Li-ion intercalation sites in the oxides.

We begin by presenting a mathematically rigorous definition of substructure and justifying this definition for use within a machine learning algorithm. We will define a similarity function between these substructures that respects both geometric and chemical similarity while avoiding computationally costly geometric rotations. We present an example of substructure clustering by investigating the lithium sites in the oxides. Finally, we apply substructural similarity to data mining the prediction of lithium insertion sites in the oxides.

4.1 Defining Substructure

In this section we develop a definition of a substructure of a crystal structure and describe a method of evaluating the similarity between two substructures. This similarity should be higher if two substructures are chemically similar and higher if two substructures are geometrically similar. We define chemical similarity by data mining from the oxide training set an ionic substitution similarity function between two ions, per Hau-

tier et al.[60,36] This function grows with the probability of the two ions substituting for each other within the same prototype within the database. Finally, we use the ionic substitution similarity as an input to the substructure similarity function, defining a similarity function via the best matching of ions in substructure s_1 to ions in substructure s_2. This similarity is a function between 0 and 1 that is continuous with respect to small geometric perturbations. Two identical substructures have similarity 1.

In the field of machine learning, the term 'feature vector selection' is used to describe the problem of finding and choosing the variables that influence the outcome of the problem at hand. The choice of feature vector should not only include all of the factors that may influence the outcome of the problem, but it should also be dense: To the greatest extent possible, it should not include information that does not influence the outcome of the problem. Feature vector selection benefits strongly from domain knowledge in the field of study.

With respect to the problem at hand, we believe that a good feature vector for substructure prediction and data mining should fulfill the following criterion.

1. The feature vector should include *geometric* information about a substructure. At a minimum, the feature vector should be able to distinguish between several commonly found coordinations such as tetrahedra and octahedra. More geometric information, for example distortions in octahedral site, or the identities of next-nearest neighbors, could also prove useful, however it is critical that such additional geometric information not create an overly sparse space. As we are expecting millions of substructures, it is also useful to minimize computational complexity by limiting feature vector size. For example, including information on the third nearest neighbors could create an overly specific feature vector.

2. The feature vector should be *continuous* with respect to small variations in geometry. As the data in this thesis is experimentally generated, we would like to create a feature vector that is robust against small perturbations in atomic position.

3. The feature vector should include *chemical* information. The strength of this thesis lies in using the ionic substitutional similarity to organize and understand

structural information; any feature vector should include the chemistries of the ions involved.

4. Lastly, the feature vector should be clearly, simply, and intuitively defined.

Motivated by the work of Villars [21], we design a feature vector based upon a substructure around a central ion. This choice allows for a clear decomposition of a given crystal structure into substructures; every ion is the center of it's own substructure. While Villars uses an atomic environment based upon Brunner and Schwarzenbach's maximum gap rule [10], in this thesis we chose to describe a feature vector based upon a weighted Voronoi polyhedron described by O'Keeffe. [71] Both methods include geometric information that is independent of symmetry and continuous against small perturbations in crystal structure.

Brunner and Schwarzenbach's maximum gap rule is based upon the radial distribution function around the central ion. The radial distribution function is normalized by r_{min}, and then the maximum gap in the radial distribution function is identified. Ions that appear with a radius smaller than the maximum gap are counted as neighboring ions; ions with a larger radius are not considered neighbors. An example histogram is shown in figure 4.1. The maximum gap rule is elegant, easily calculable, and continuous. However, the clarity of this method breaks down when a maximum gap is not clearly discernable.

O'Keeffe's method is based upon the Voronoi decomposition of a crystal structure. [71] A crystal structure c can be decomposed into a set of Voronoi polyhedra with one central ion x inside each polyhedron. [91] Each polyhedron represents the set of points that are closer to x than any other ion. Each face of the polyhedron surrounding x is generated by the plane bisecting the line between x and a neighboring ion y, and subtends a solid angle Ω_y from the central ion. Following the work of O'Keeffe, for a given polyhedron, the neighbor y corresponding to the greatest solid angle is assigned a weight $w_y = w_{max} = 1$, and every other neighbor z is assigned a weight $w_z = \Omega_z/\Omega_{max}$.

Neighboring ions with greater Voronoi weights tend to be closer to the central ion i; they also tend to have fewer nearby neighbors within the same solid angle from the central ion. Correspondingly, neighbors with small Voronoi weights tend to be further away from the central ion i, and tend to have more nearby neighbors. O'Keeffe's method is also elegant, continuous, and calculable with Barber's elegant Quickhull

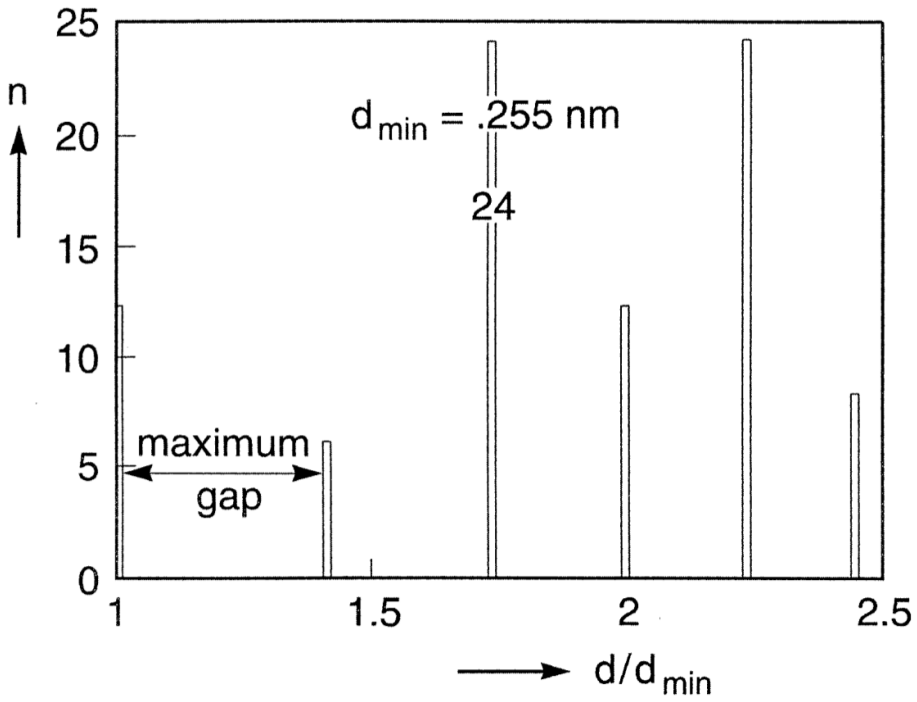


Figure 4.1: A radial distribution function based next-neighbor histogram for the FCC Cu crystal structure.[21]

algorithm.[6] Additionally, O'Keeffe's algorithm is intuitive and rigorously mathematically defined.

For the purposes of this chapter, we represent the substructure s of a central ion i in a crystal structure c to with the following data structure:

- We identify the central ion and it's ionic species i
- We keep an unordered list of neighboring ions, represented by (ionic species x and Voronoi weight w_x) pairs, $\{(x, w_x)\}$.

For example, a phosphate tetrahedron would be stored thus:

- Central ion: P^{4+}
- Peripheral ions: $(O^{2-}, 1)$, $(O^{2-}, 1)$, $(O^{2-}, 1)$, $(O^{2-}, 1)$

The choice to record O'Keeffe's Voronoi polyhedra weights instead of the distance from a neighboring ion to the central ion was also motivated by the desire to create an informationally dense feature vector. By keeping an unordered list of neighboring ions with associated weights instead of all the geometric information about the substructure, we have forfeited a large amount of angularly dependent geometric information. Keeping all the geometric information about the substructure would require costly geometric rotations when comparing two substructures to each other. However, the information captured by O'Keeffe's weighted Voronoi polyhedra contains a reasonable mixture of angular and radial information. Not only do neighbors that are closer to the central ion have higher weights, but neighbors that are farther from other neighbors also have higher weights.

This feature vector choice allows us to represent the atomic environment of an ion in a chemistry and geometrically sensitive manner. Furthermore, the feature vector will remain unchanged if the crystal structure is rescaled by a constant, allowing us to robustly compare the atomic environments of ions of differing radii. We note that the proposed definition of a substructure is mathematically rigorous, in that it can be calculated for every crystal structure. It is not dependent of symmetry, and it is continuous against small variations in crystal structure, with respect to both small movements of ions and small changes in lattice vector.

4.2 Similarity between Substructures

In this section we develop a similarity function between two substructures. This function should be higher if two substructures are chemically similar and higher if two substructures are geometrically similar. This function should be 1 if two substructures are identical, and should always be greater than 0.

Two substructure s_i and s_j have central ions i and j and sets of neighbors N_i and N_j. Each neighboring ion $x \in N_i$ and $y \in N_j$ has an associated weight w_x or w_y that satisfies $0 \leq w \leq 1$.

We begin be defining a score between two neighboring ions x and y:

$$\text{Score}(x, y) = \text{Sim}_{\text{ion}}(x, y) \min(w_x, w_y) e^{\frac{-(w_x - w_y)^2}{c^2}} \tag{4.1}$$

This score satisfies the following properties:

- it is greater if the two ions are chemically similar, due to the contribution of the ionic substitution similarity function.
- It is greater if the weights of the two ions are higher, due to the contribution of the minimum of the two weights, and
- it is greater if the weights of the two ions are close to each other, due to the contribution of the Gaussian function.

The parameter c allows the user to tune the sensitivity with which the score penalizes different weights. A higher value of c yields a wider spread in the Gaussian, allowing for greater differences between the weights and lesser geometric sensitivity.

This score represents the similarity of the two neighboring ions to each other, taking into account both chemical and geometric similarity. It tends to be higher if the neighboring ions are more central to their respective substructures.

Next, we define a product between two substructures:

$$\text{Product}(s_i, s_j) = \max_{\text{all matchings}} \Sigma_{x,y \in \text{matching}} \text{Score}(x, y) \tag{4.2}$$

where the sum is taken over the pairing of ions $x \in N_i$ to ions $y \in N_j$ that maximizes the product. If there are more ions in one substructure than the other, the excess

ions in the larger substructure will remain unpaired, and will not contribute to the sum. This product between two substructures does not take into account the similarity of the central ion, and is used throughout this chapter as the application under consideration is the identification of Li sites.

An alternative product between two substructures that takes into account the similarity of the central ions i and j is given by:

$$\text{Product}(s_i, s_j) = \text{Sim}_{\text{ion}}(i, j) + \max_{\text{all matchings}} \Sigma_{x,y \in \text{matching}} \text{Score}(x, y) \quad (4.3)$$

This product should be used when comparing substructures with differing central ions, as in chapter 5.

Finally, we normalize the product of two substructures to obtain the substructure similarity function. This type of normalization is necessary to avoid assigning larger substructures larger similarities due to their greater number of neighboring ions.

$$\text{Sim}_{\text{substruct}}(s_i, s_j) = \frac{\text{Product}(s_i, s_j)}{\sqrt{\text{Product}(s_i, s_i), \text{Product}(s_j, s_j)}} \quad (4.4)$$

We define substructural similarity to be the resultant similarity function.

It is worth noting that discarding the excess unpaired neighboring ions in the maximum matching represents large computational savings over the maximum matching calculation in the composition similarity chapter. When calculating composition similarity, it is physically intuitive to compare the same number of ions from one composition to the other; for this reason the maximum matching is calculated over the lowest common multiple of the number of ions in each composition. It is not uncommon for this maximum matching to be calculated over 50-80 ions. However, when calculating similarity between substructures, the geometry of the substructures at hand require that we only compare the number of ions in one substructure to the number of ions in the other. No lowest common multiple is required. On average, a Voronoi polyhedron in the oxide data set will have 16 neighbors. Recalling from chapter 3 that the Hungarian algorithm operates in $\mathcal{O}(n^3)$ time, this represents substantial computational savings.

4.3 Application to Li Site Prediction

We now have the ability to parse crystal structures into substructures and to organize those substructures by similarity to each other. To demonstrate the capabilities of this set of tools, we use substructural decomposition and similarity to analyze the set of Li sites in the oxides. We will begin by describing the oxide data set at hand, and then we will predict Li sites for a set of oxide crystal structures.

4.3.1 Data Set Construction

The data set that we used to validate substructure similarity was very similar to the data set used to calculate composition similarity. The rationale behind selecting the oxides can be read about in chapter 3.2.1. The oxide data set constructed for the work in this section differs from the one used in the composition similarity chapter in that the compounds were additionally screened for charge-balanced structures only; the ICSD database was updated to include all data from year 2014, and the affine mapping algorithm used to prototype the database was updated to the most recent pymatgen[73] version 2.1.2.

All of the compounds in the Inorganic Crystal Structure Database[7] (ICSD) 2012 were searched for compounds that satisfied the following criteria:

- Compounds must be oxides, as indicated by at least 20% oxygen content by ion count.
- Compounds must not be peroxides or superoxides, as indicated by O-O bond lengths L < 1.50 Å.
- Compounds must not be marked 'high pressure', 'HP', 'high temperature', or 'HT.'
- Compounds must not have improbably short (< 1Å) bond lengths.
- Compounds must not have a mismatch between the reported composition and the ions given in the crystal structure.
- Compounds must not contain hydrogen. The reported crystal structures of compounds containing hydrogen are often unreliable.

- Compounds must be charge balanced, as indicated by the total charge of all the species reported summing to an absolute value $< |0.001|$.

The resultant oxides were sorted into structure prototypes using an elegeant affine mapping algorithm by Stephen Dacek and Shyue-Ping Ong.[40,11] The data set was further cleaned by removing duplicates, defined as compounds with the same composition and the same structure prototype, resulting in a final data set of 5,509 oxide compounds.

After cleaning, the complete data set was randomly split into 10 equally sized subsets for cross validation. Each subset served in turn as a test set, while the other 9 subsets served as the training set for cross validation. The training set, comprising 90% of the compounds, represents the database of known compounds to be data mined. The remaining 10%, called the test set, mimics a set of as-yet-unseen compounds which we use to evaluate the efficacy of our site prediction algorithm.

4.3.2 Site Prediction Algorithm

In the following section, we apply substructural similarity to to the prediction of Li sites in the oxides. While there are many possible ways to use the ionic, composition, and substructural similarities we have constructed to predict Li sites, we describe below one simple method that validates the substructural similarity.

The ionic substitution similarity functions was extracted from the training set as described in chapter 2. As the substructural similarity function balances chemical similarity versus geometric similarity, we used the tunable ionic similarity function given by equation 2.10. The parameters σ and μ, which were used in the tunable ionic similarity, and the parameter c, used in the substructural similarity, were set using nested cross-validation. Each training set was further subdivided into 5 partitions; each partition was held apart as a test set in turn, creating 5 internal cross-validation sets for each external cross-validation set. The algorithm described below was run for each internal cross validation set while we varied each parameter σ, μ, and c in turn; the set of parameters that yielded the highest overall area under curve score (described below) was chosen. This optimal set of parameters $\sigma = 0.3, \mu = 0.7$, and $c = 0.05$ was then used to generate the results in the external cross-validation loop.

Each compound in the training set was searched for Li sites, and a database of Li substructures was compiled. The ionic substitution similarity and the database of Li substructures consist of all the information extracted from the training set.

For each Li-containing compound in the test set, the Li sites were removed. The resultant, Li-free crystal structures represent delithiated oxides for which we will predict the Li sites. The Voronoi polyhedra for the Li-free crystal structures were computed using the quickhull algorithm for convex hulls by Barber et al.[6] The set of points given by the corners of each Voronoi polyhedron and the centers of each Voronoi polyhedra face constitute a reasonable set of potential lithiation sites; we call this set of points the Voronoi points of the crystal structure. The Voronoi polyhedron corners represent the set of points that are equidistant from their four closest neighbors, and thus represent the set of points that are as far away as possible from any other point.[91] Voronoi polyhedron face centers are another common site for inserted species. Finally, all sites and Voronoi points for each Li-containing compound were grouped together into symmetrically identical sites using pyspglib[85] to reduce computational complexity.

We attempted to rediscover the removed Li sites in each Li-containing compound in the test set. For each symmetrically distinct Voronoi point in a crystal structure, we calculated the substructural similarity between this Voronoi point and every known Li substructure obtained from our training set. We ranked each symmetrically distinct Voronoi point by its distance to the most similar Li containing substructure, where distance is given by $1 -$ similarity, to produce an ordered list. Going down the list, we guessed Li sites subject to the following rules:

1. If the current symmetrically distinct Voronoi point is within $r < 0.5$ Åof an undiscovered Li site, it is considered a correct guess. That Li site and all of its symmetrically distinct neighbors are now marked as discovered, and cannot be discovered again.

2. If the current Voronoi point is within $r < 0.25$ Åof a previously guessed Voronoi point, this Voronoi point is not guessed. This rule is necessary because Voronoi points are commonly found clustered together.

As in chapter 3, we found it useful to compare the performance of the substructural similarity ranking versus another ranking method. One simple and interesting ranking

method involves ranking the sites by radius, where the radius is given by the site-to-site distance to the nearest neighbor. The average Li site radius in the oxide database was 2.1 Å. The following section gives the results for ranking both via the substructural similarity method, and by distance $d = |r - 2.1|$, where r is the site radius given by the distance from the center of the site to the center of the closest ion.

4.3.3 Results

Figure 4.2 depicts the results of Li site prediction in the oxides by the two ranking methods given above. The x axis shows the probability of achieving a given result for a specific oxide selected from the data set; the y axis shows the number of wrong guesses necessary before finding all the Li sites. The data shown is aggregated across all 10 cross-validated sets, and thus represents Li site prediction across all unique 302 Li-containing oxides.

Ranking potential Li sites by substructural similarity fares better than ranking by site radius, consistently requiring 2-3 times fewer incorrect guesses. Substructural similarity finds all the Li sites within a crystal structure with 9 incorrect guesses 90% of the time. 50% of the time, it finds all the Li sites without any incorrect guesses. However, on the most difficult 5%, it requires over 50 incorrect guesses to find all the Li sites; this is because approximately 5% of Li-ion sites are not within 0.5 Åof a Voronoi point, and thus cannot be found by this algorithm. Ranking by substructural similarity would be a potentially useful first screening mechanism in a high-throughput search for Li sites.

Figure 4.3 depicts the results of Li site prediction via a receiver operating characteristic (ROC) curve. A ROC curve depicts the true positive fraction versus the false positive fraction for a binary classifier as the predictive threshold is varied. ROC curves are commonly used to assess the tradeoff between sensitivity and specificity[34]. In this case the binary classifiers at hand are classifying Voronoi points as Li sites or not Li sites; the true positive fraction or the sensitivity is the fraction of correctly identified Li sites, and the false positive fraction is the fraction of incorrectly identified not Li sites. The ratio of true positives to false positives changes as we vary the distance below which a given Voronoi point is classified as a Li site.

A perfect classifier should correctly identify all of the true positives before returning a false positive. The ROC curve of a perfect classifier would go straight up from (0, 0)

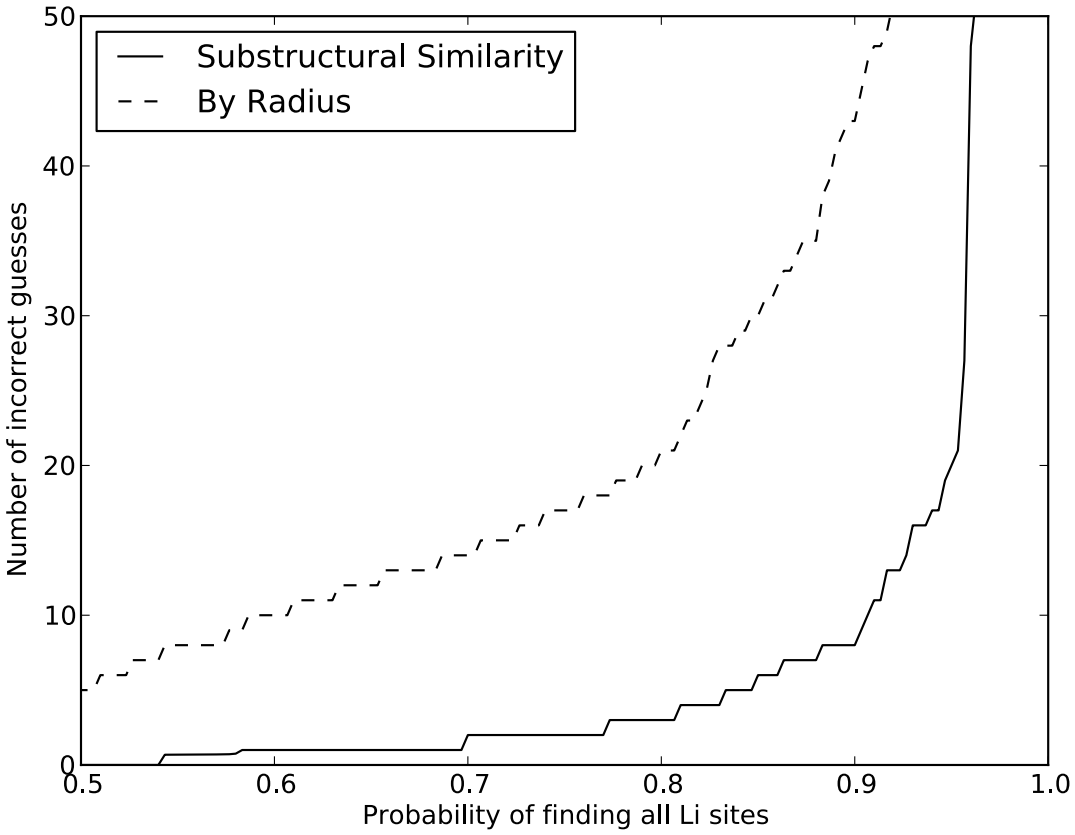


Figure 4.2: The Li site identification rate. The solid line shows the number of incorrect guesses before identifying all the Li sites in a structure via the substructural similarity method; the dotted line shows the number of incorrect guesses using the radius of the site. Substructural similarity finds all the Li sites within a crystal structure with 9 incorrect guesses 90% of the time.

to (0, 1) before going right to (1, 1). The area under the ROC curve (AUC) is a commonly used figure of merit to assess the quality of a binary classifier. A perfect classifier has AUC = 1.

Examining figure 4.3, the benefits of Li site prediction by substructural similarity becomes clear. Substructural similarity achieves a higher area under the curve by correctly identifying Li sites earlier, before mis-identifying non-Li sites.

Finally, figure 4.4 depicts the performance of Li site classification broken down by compound complexity. Here, we define the compound complexity as the number of symmetrically distinct sites in the crystal structure; this number appears to be linearly correlated with the number of symmetrically distinct Voronoi points in the crystal structure (shown in black). Again, the number of guesses required to find all the Li sites by substructural similarity (shown in blue) is consistently lower than the number of guesses required to find all the Li sites by radius (shown in red). Interestingly, the number of guesses required to find all the Li sites does not appear to be correlated with the complexity of the compound. Finally, there are a number of outlying poor performers distributed across several compound complexities which require 100 or more guesses to identify all the Li sites.

4.3.4 Discussion

We have developed a definition of substructure that is mathematically precise, continuous against small variations in position, and calculable for every atomic position in every crystal structure. We built upon that definition of substructure a similarity function that takes into account geometric and chemical similarity and produces a number between 0 and 1 that is higher if the two substructures are more similar. We used this definition of substructure and substructural similarity to predict Li sites in oxide compounds, finding all the Li sites within a crystal structure with 9 incorrect guesses 90% of the time. This application has the potential to greatly reduce computational time in the search for Li insertion sites, as it is not uncommon for there to be hundreds of symmetrically distinct Voronoi points per oxide crystal structure.

One of the strengths of the proposed data mining algorithm for interstitial insertion sites is that it is both general and flexible. While the presented work considers only the insertion of Li ions, there is no reason this work could not be extended to predict

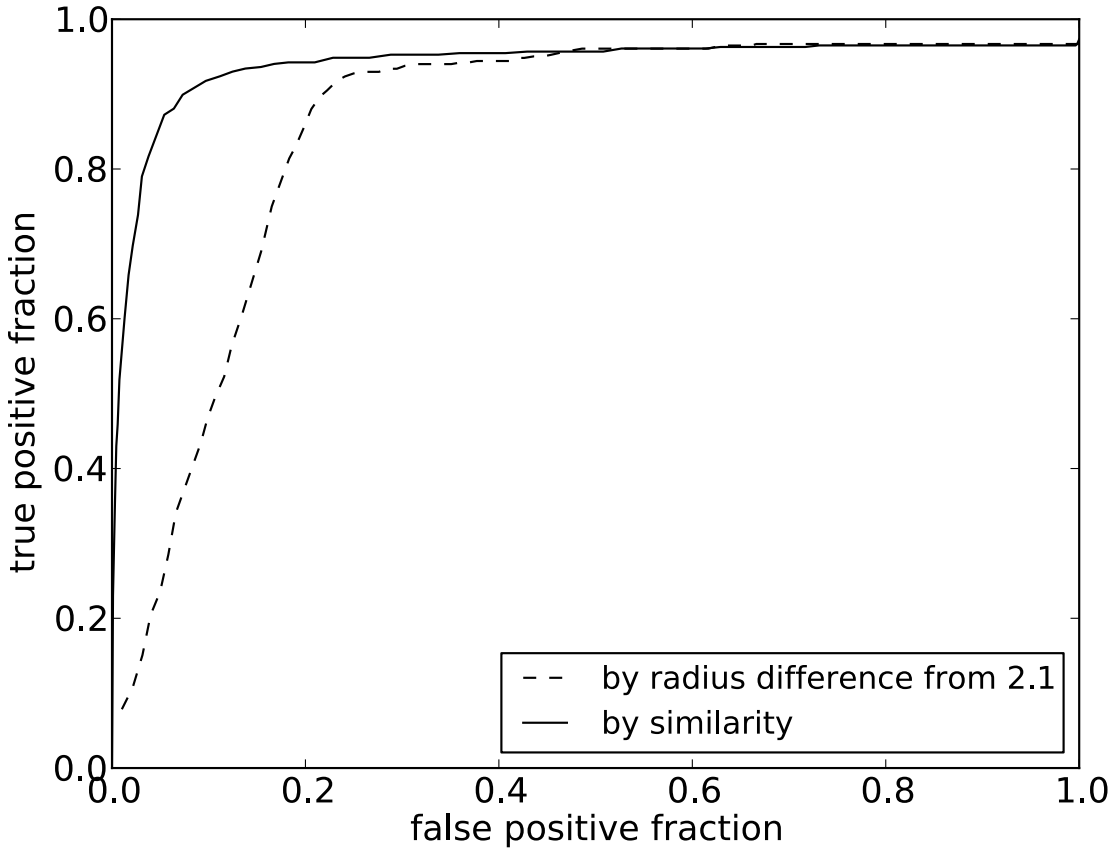


Figure 4.3: The receiver operating characteristic for Li site classification. The x axis depicts the false positive fraction, or fraction of non-Li-site Voronoi points that were mistakenly identified as Li sites. The y axis depicts the true positive fraction, or the fraction of Li sites that were correctly identified as Li sites. The black line represents classification by substructural similarity; the dotted line represents classification by site radius.

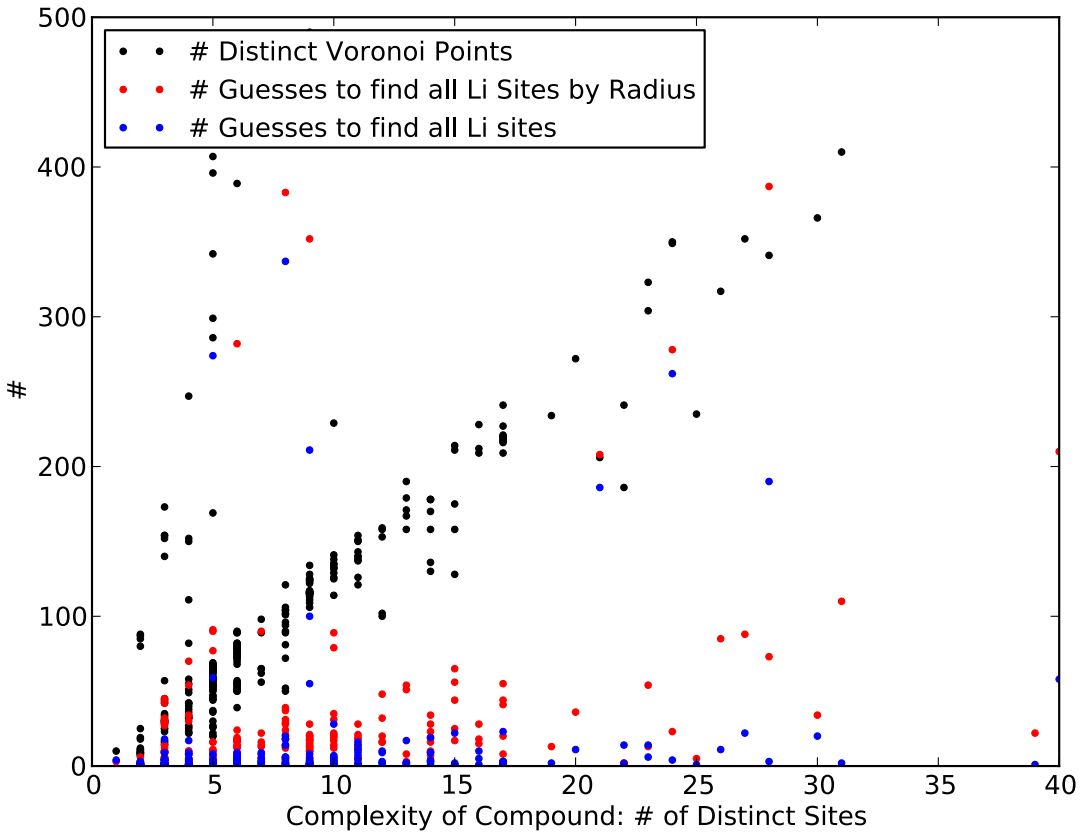


Figure 4.4: Performance of Li site prediction by compound complexity. The number of incorrect guesses required to find all the Li sites is plotted versus the complexity of the oxide host. The complexity of the host is given by the number of symmetrically distinct sites in the host structure. Ranking by substructural similarity performs equally well on simple oxide structures as it does on more complex oxides.

the insertion sites for Na or Mg ions; indeed, this framework can be used to identify, analyze, and predict the sites of any ion.

Furthermore, the proposed data mining algorithm was used to validate the substructure and substructural similarity constructions. We present a brief analysis of the failure modes of this data mining algorithm. Firstly, approximately 5% of Li sites are never found because 5% of the Li sites are not within 0.5 Åof a Voronoi point. Next, of 312 lithiated oxides, there were 8 structures for which the Li site finding algorithm required more than 100 guesses to identify all the Voronoi points that were within 0.5 Åof a Li site. Table 4.1 gives summary information for the those 7 lithiated structures. The table shows the number of symmetrically distinct Li sties, the number of guesses to find all the Li sites, and the number of Voronoi Points in the crystal structure. Finally, the table also gives two other quantities of interest. The minimum distance between any Voronoi point in the crystal structure and any Li site in the training set is sometimes pertinent; the average distance between any voronoi point and any Li site is approximately 0.6. Therefore if the *minimum* distance between any Voronoi point in the structure and any Li site is higher than 0.9, this crystal structure would be a statistical outlier and should be further examiend. Secondly, the distance at which the last Li site was identified is pertinent as a measure of how unusual the most unusual Li site in the crystal structure is. The greater the distance, the more unusual the site. Table 4.1 gives all of the quantities described above.

Looking through Table 4.1 , we notice in the last two columns two structures with unusually high Voronoi distances. $Li_1Nd_9Mo_{16}O_{35}$ and $Li_8Rb_8B_{32}O_{56}$ were both reported with no charge state information. When ions are reported without charge state information, the algorithm searches for similar ions with a charge state of 0, yielding very low similarities to known substructures. The minimum distances between *any* Voronoi point and all known Li sites for these two structures are 0.92 and 0.95. Error due to unreported charge states can be easily addressed in future work by screening out such crystal structures.

The other 6 structures illustrate a weakness of any data mining algorithm, in that the data mined predictions are only as good as the data set at hand. The distances between the last found Li site and the closest Li site in the training set for each of these structures is greater 0.44, whereas the average distance between an Li site and the closest Li site in the training set is 0.31. In contrast, the average distance between a ran-

Table 4.1: Failure Modes for Li Site Prediction

Composition	Li Sites	Guesses	Voronoi Points	Minimum Distance	Maximum Li Distance
$Li_3^+V_2^{3+}(P^{5+}O_4^{2-})_3$	2	117	221	0.36	0.51
$Li_1Nd_9Mo_{16}O_{35}$	1	186	206	0.92	0.99
$Li_{56}^+Si_{14}^{4+}O_{56}^{2-}$	19	190	909	0.33	0.44
$Li_{18}^+Cr_6^{3+}P_{16}^{5+}O_{58}^{2-}$	3	211	1049	0.31	0.44
$Li_{10}^+As_{22}^{5+}U_{26}^{6+}O_{138}^{2-}$	5	236	1220	0.33	0.44
$Li_8Rb_8B_{32}O_{56}$	2	262	349	0.95	0.99
$Li_{12}^+W_6^{6+}O_{24}^{2-}$	2	274	396	0.36	0.54
$Li_{10}^+Cl_2^-B_{14}^{3+}O_{25}^{2-}$	2	357	615	0.41	0.60

domly drawn Voronoi point and the closest Li site in the training set is 0.57. The distribution of Li site distances and the number of guesses necessary to find an Li site is given in figure 4.5. The red line and the right hand axis plot the distance of a site to the closest Li site in the training set versus the average number of guesses necessary to find it; the blue line and the left hand axis plot the distance of a site to the closest Li site in the training set versus the percentage of Li sites in the test set that are found at that distance. A good prediction algorithm increases the lag between the blue curve and the red curve, identifying all of the Li sites before increasing the number of guesses. The other 6 structures represent the tail end of the red curve; the unusually high distances of the last Li sites to be found indicate that there are no highly similar Li sites in the training set. This is either because of an usual chemistry as demonstrated by $Li_{10}^+As_{22}^{5+}U_{26}^{6+}O_{138}^{2-}$, or an unusual geometry.

While the substructural data mining methods developed in this chapter are general and could theoretically be applied to any number of ionic species, the weakness of data mining lies in the quality of the data set at hand. For example, the quality of the Li site predictions depends strongly on the number of Li sites in the database; in this case, the oxide database provided 1631 occurrences of Li in 312 crystal structures across 458 unique substructures. However, if we were to repeat this prediction procedure for Mg site predictions, the same oxide database would provide only 839 occurrences of Mg in 140 crystal structures across 176 unique substructures. For this reason, we expect data mining predictions to fare less well when applied to less common chemistries. In

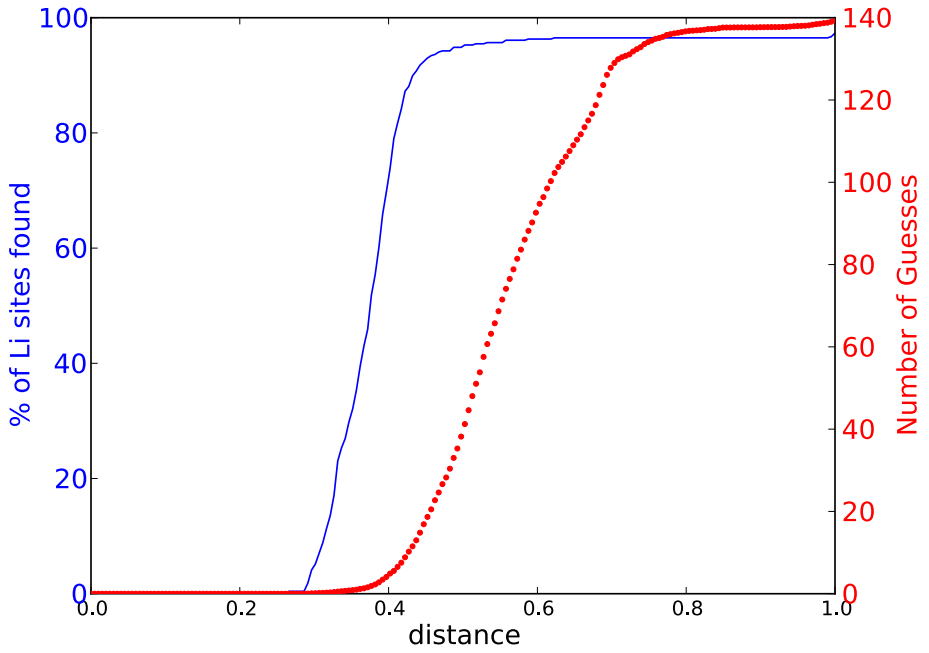


Figure 4.5: Distance distribution of Li sites and number of guesses. The x axis depicts the distance as calculated by the similarity metric between an Li site in the test set and the closest Li site in the training set. The blue y axis on the left hand side depicts the distribution of Li sites versus distance; the red y axis on the right hand side depicts the average number of guesses at a given distance. An ideal ranking system would increase the lag between the blue and the red lines, finding 100% of the Li sites before requiring more than 1 guess.

such a scenario, it would be reasonable to use ionic similarity to gather data from across similar chemistries. We could include data from ions with high ionic substitutional similarity (like Ca and Sr), weighted by ionic substitutional similarity, when making predictions for Mg sites.

There are several reasonable extensions and permutations of the current algorithm. We extracted a list of known Li sites and compared potential sites directly to the known sites, ranking potential sites by similarity to a known site. It would be reasonable to take into account other factors - for instance, the radius of the potential site, the ratio of anions to cations in the host structure, or even the composition of the host structure. One could easily extend the algorithm by conditioning the ranking of the potential site upon not only substructural similarity to an Li site, but also substructural similarity to a known Na site. One could also condition the ranking upon the composition of the host structure being similar to the composition of a structure that is known to host Li. Another algorithm would search through the database for structures with similar compositions and extract Li sites from those compositions only. There are many possibilities for reasonable site prediction algorithms, each with its own tradeoffs in terms of computational time, dependency on the robustness of the data set at hand, and quality of potential results.

We explore the possibility of combining radius information with the substructural similarity. Figure 4.6 shows the Voronoi points (blue) and Li sites (red) from 30 randomly selected Li-containing oxides. The x-axis depicts the similarity versus the most similar Li site in the training set; the y axis depicts the radius difference from 2.1 Å. The Li sites are clearly clustered in the lower left-hand corner. However, it is not clear from this graph to what extent the Li sites ares separable from the Voronoi points, as there are also many Voronoi points in the lower left-hand corner.

We attempted to combine radius and substructural information via a variety of linear regression techniques. We trained a linear classifier on the set of Voronoi points from the training set; each Voronoi point contributed both radial and substructural similarity distances, and information on whether or not it was within 0.5 Åof a Li site. We then used the linear classifier to classify points in the test set; figure 4.7 shows the cumulative results of ridge regression over all 10 cross-validated sets. Linear regression does not improve upon the predictions made by substructural similarity.

Examining the ROC curve shown in figures 4.7 and 4.3 more closely, we find that

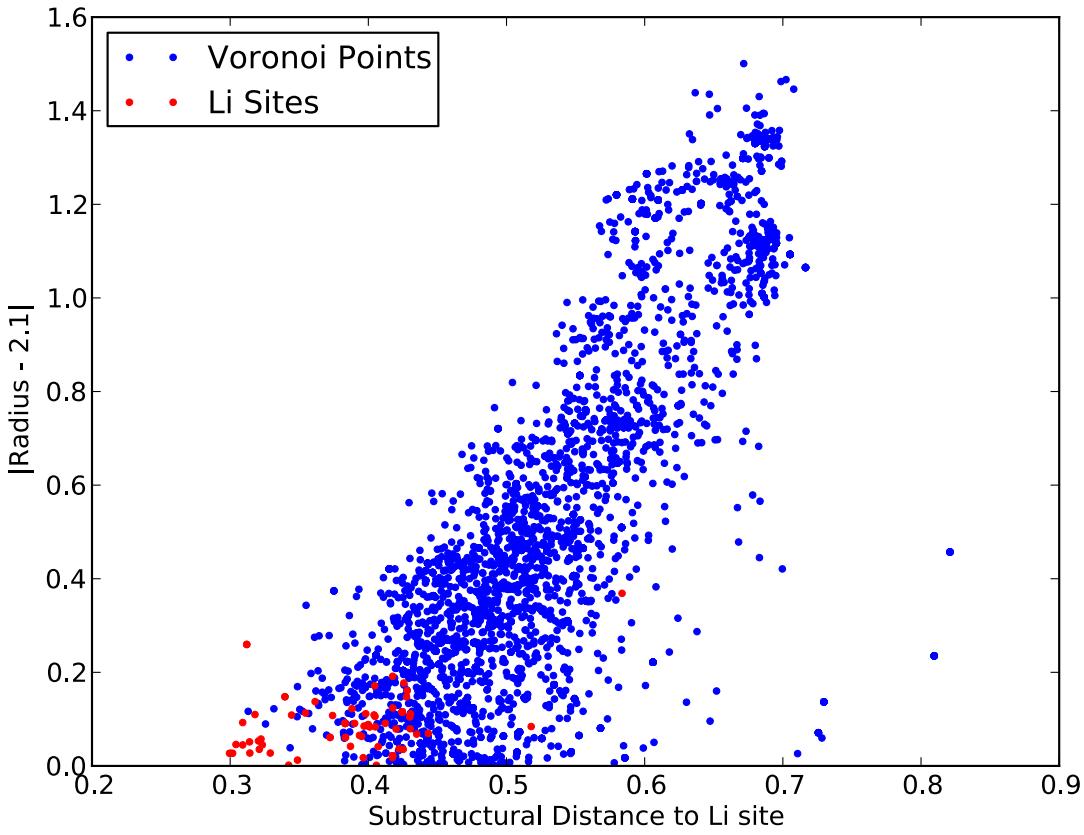


Figure 4.6: Li sites and Voronoi points from 30 randomly selected Li-containing oxides. While the Li sites are clearly clustered in the lower left-hand corner, it is not obvious to what extent the Li sites are separable from the Voronoi points.

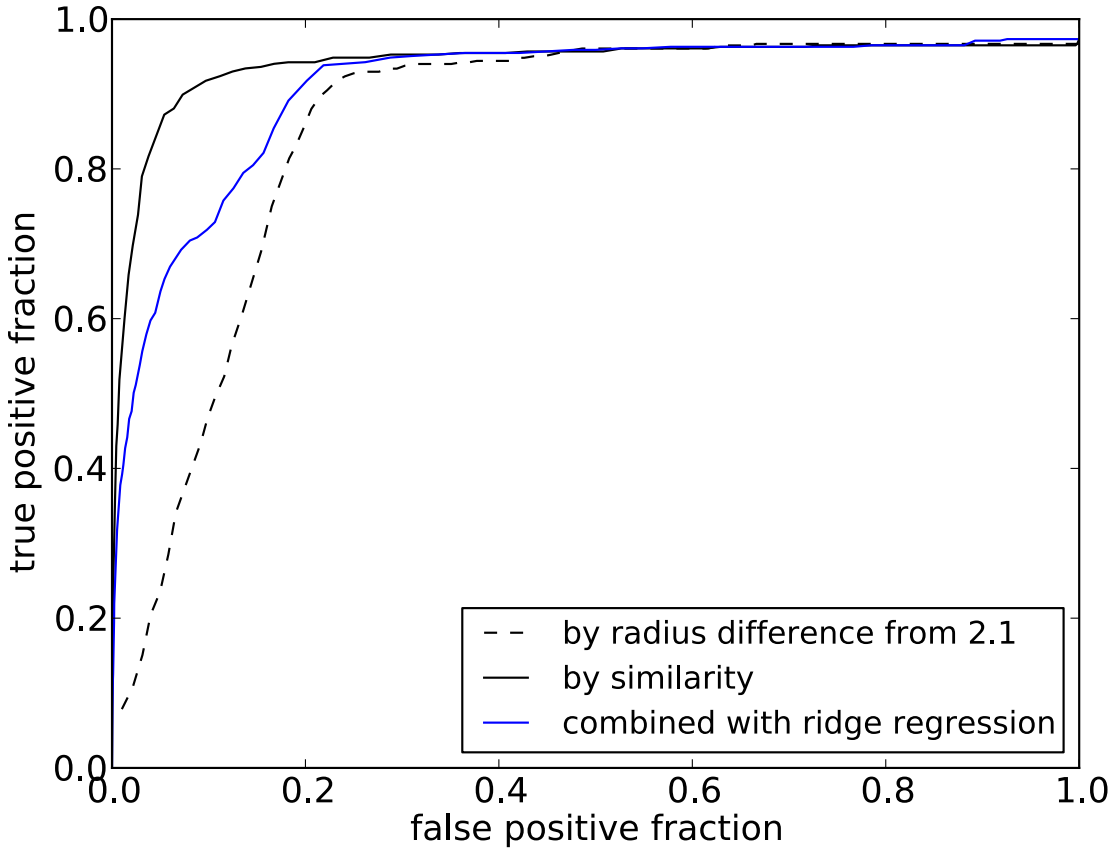


Figure 4.7: The receiver operating characteristic curve is shown in blue for a ridge regression model based upon both radius and substructural similarity information. The blue curve has a lower area under curve than ranking by substructural similarity alone.

all curves reach their maximum true positive rate at around 95%. Approximately 5% of the Li sites are never found because 5% of the Li sites are not within 0.5 Åof a Voronoi point. A reasonable extension of this work would include a more thorough search for potential Li sites; it is not clear that the Voronoi points are an optimal set. From figure 4.4, we infer that the number of incorrect guesses required by the substructural similarity algorithm does not grow with the number of Voronoi points, so the performance cost of adding more potential Li sites is minimal. The cost of computing the substructural similarly low and is easily parallelized. One could consider discretizing the space within a given crystal structure into 0.5 Åcubes and reducing the resultant points by symmetry.

It is worth mentioning that while the framework presented in this chapter captures only local, first-neighbor interactions, there are a number of potentially meaningful extensions to explore. It would not be difficult to extend the substructures in this work to include second-neighbor interactions by representing each substructure as a graph that includes second-neighbor connections. Alternatively, one can view each crystal structure as an overlapping tiling of substructures; each ion participates not only in the substructure to which it is central, but also in all of it's first neighbor substructures. Using this framework, one can mine for patterns in the interconnectivity of substructures. What substructures tend to overlap the most? What combination of substructures allows for Li-ion diffusion?

We have outlined a mechanism for the prediction of Li sites in the oxides. For the purpose of validation, we began with lithiated oxide crystal structures, removed the Li ions, and then predicted the Li sites in the artificially delithiated structures. This construction allowed us to obtain an experimentally verified set of Li sites. However, when applying this algorithm to the identification of Li sites in delithiated structures, we expect to obtain less accurate results for two reasons. Firstly, structures relax when Li ions are added or removed. The artificially delithiated structures we ran our predictions on were not relaxed; in essence, they were artificially frozen in a structure with Li-ion vacancies, making it easier to identify Li-ion sites. Secondly, this algorithm does not take into account the effect of Li concentration on the Li sites predicted. In effect, this algorithm is a mechanism for the prediction of Li sites in the dilute limit. It would be theoretically possible to insert Li ion-by-ion into a structure, re-running the prediction algorithm between each insertion to find the next Li site. However, as this algorithm

only takes into account local effects, we expect the ability of this algorithm to predict Li orderings to be limited.

We have used a definition of substructural similarity that can be tuned for greater or lesser geometric and chemical sensitivities. We set the tuning parameters σ, μ, and c to maximize the area under the ROC curve for the application of predicting Li sites in the oxides. However, differing applications of the substructural similarity function will call for different tunings. For example, the current algorithm ranks all potential Li sites across a host of candidate oxide compounds. Another potential application of substructural similarity would be to rank potential Li sites within a single oxide compound to predict diffusion pathways. In this second application, we would expect the optimal tuning of the substructural similarity to be more sensitive to geometric differences and less sensitive to chemistry, as finding a diffusion path requires the ability to distinguish between small geometric differences.

This chapter has presented a definition of substructure that is mathematically rigorous, continuous with respect to small displacements in ion position, and dependent upon both chemistry and geometry. We have further defined a similarity function between any two substructures that is 1 if the two substructures are identical, and decays towards 0 with growing differences in geometry and chemistry. This definition of substructure and substructural similarity allow for the decomposition and analysis of crystal structure databases. We have validated substructural analysis by predicting Li sites in the oxides.

5

Conclusion

Thus far we have presented similarity functions between ions, compositions, and substructures. We have shown how these similarity functions preserve similarity in crystal structure and can be used to predict structure prototype and Li insertion sites. In this chapter we present an extension of this work in the form of a similarity function between crystal structures. We discuss the potential of this work to aid in the prediction of entirely new crystal structures, and conclude with a few thoughts on future applications.

5.1 Structural Similarity

The composition and substructural similarity functions are both functions of ionic similarity that apply maximum matching algorithms to find the most similar mapping of ions. A straightforward extension of these ideas leads to a similarity function

between crystal structures.

A crystal structure x with m atomic sites can be decomposed into m substructures $\{s_1, s_2, ...s_m\}$ as described in chapter 4. Viewing each crystal structure as a *composition* of substructures, we express each crystal structure as a set of substructures $\{s\}$ together with the number of times each substructure appears $\{n\}$. We reduce each crystal structure such that the reduced version has the smallest integers $\{n\}$ that preserve the correct ratios between the substructures. The sum Σn of the number of substructures in the reduced composition is the total number of substructures n_{total} of a given reduced crystal structure.

Given two crystal structures x_1 and x_2, we find the lowest common multiple n_{lcm} of n^1_{total} and n^2_{total}. Two sets of substructures $\{s_1\}$ and $\{s_2\}$ of length n_{lcm} are created by enumerating the substructures of x_1 and x_2 the appropriate number of times. Searching through all the possible matchings (ss_1, ss_2) of the substructures ss_1 in $\{s_1\}$ to the substructures ss_2 in $\{s_2\}$, the matching that maximizes the average similarity of the two sets is found. This maximal average similarity is defined as the crystal structure similarity between x_1 and x_2.

$$\text{Sim}_{\text{structure}}(x_1, x_2) = \max_{\text{all matchings}} \frac{\Sigma_{ss_1, ss_2 \in \text{matching}} \text{Sim}_{\text{substruct}}(ss_1, ss_2)}{n_{lcm}} \tag{5.1}$$

The crystal structure similarity yields a rating between 0 and 1 for every pair of crystal structures x_1 and x_2, with identical compositions having similarity 1. Based on data mined values for the probability with which each ion will substitute for another within the same prototype, this crystal structure similarity provides a quantitative, local substructure based distance semi-metric between crystal structures that is sensitive to both chemistry and geometry.

5.2 Future Directions

We've presented a framework for the computation of data mined similarities and distances between ions, compositions, substructures and crystal structures. Compounds with high composition similarity are likely to have high structural and substructural similarity. The similarities between substructures and crystal structures are continuous

against small displacements in position and invariant under rotation and translation. In this section we discuss extensions and potential applications of this work.

5.2.1 Negative Data

One weakness of the work presented in this thesis is that it is based solely upon positive data. The ionic substitutional similarities that form the basis of this work are trained upon an experimentally determined database of stable crystal structures. If a crystal structure is absent from this database, it is impossible to distinguish whether it is absent because it is structurally unstable or if it is absent because it has not yet been sampled. This absence of negative data extends to every prediction made in this thesis; while we can observe that certain predictions are *likely*, it is impossible for us to say whether other predictions are *unlikely* or simply the result of absent data.

One method by which this weakness could be addressed would involve a database of *ab initio* calculations. The careful construction of such a database would calculate the formation energies of a number of highly unstable phases, allowing for the collection of data on unlikely substitutions. This extension of this work would enable predictions of which compositions are *unlikely* to form in a given prototype, and which ions are *unlikely* to inhabit particular sites.

5.2.2 Organization of Data

As mentioned previously in Chapter 2, the creation of a similarity function and the associated distance function imposes a topology upon the space, creating a natural means by which to cluster and organize ions, compositions, substructures and crystal structures. This clustering can be used to create a hierarchy of compositions, like the one shown in figure 3.8, allowing for the grouping of chemically similar compositions. Alternatively, we can create a hierarchy of crystal structures. Figure 5.1 depicts the clustering of 25 randomly selected oxides by composition similarity and structural similarity. Oxides are connected by horizontal lines at the height of the distance between them (depicted by the y axis); similar oxides are connected by lower horizontal lines.

Clustering by composition similarity yields very similar results to clustering by structural similarity, as one would expect. We have shown previously that compounds with similar compositions are more likely to have the same structure prototype, and the two

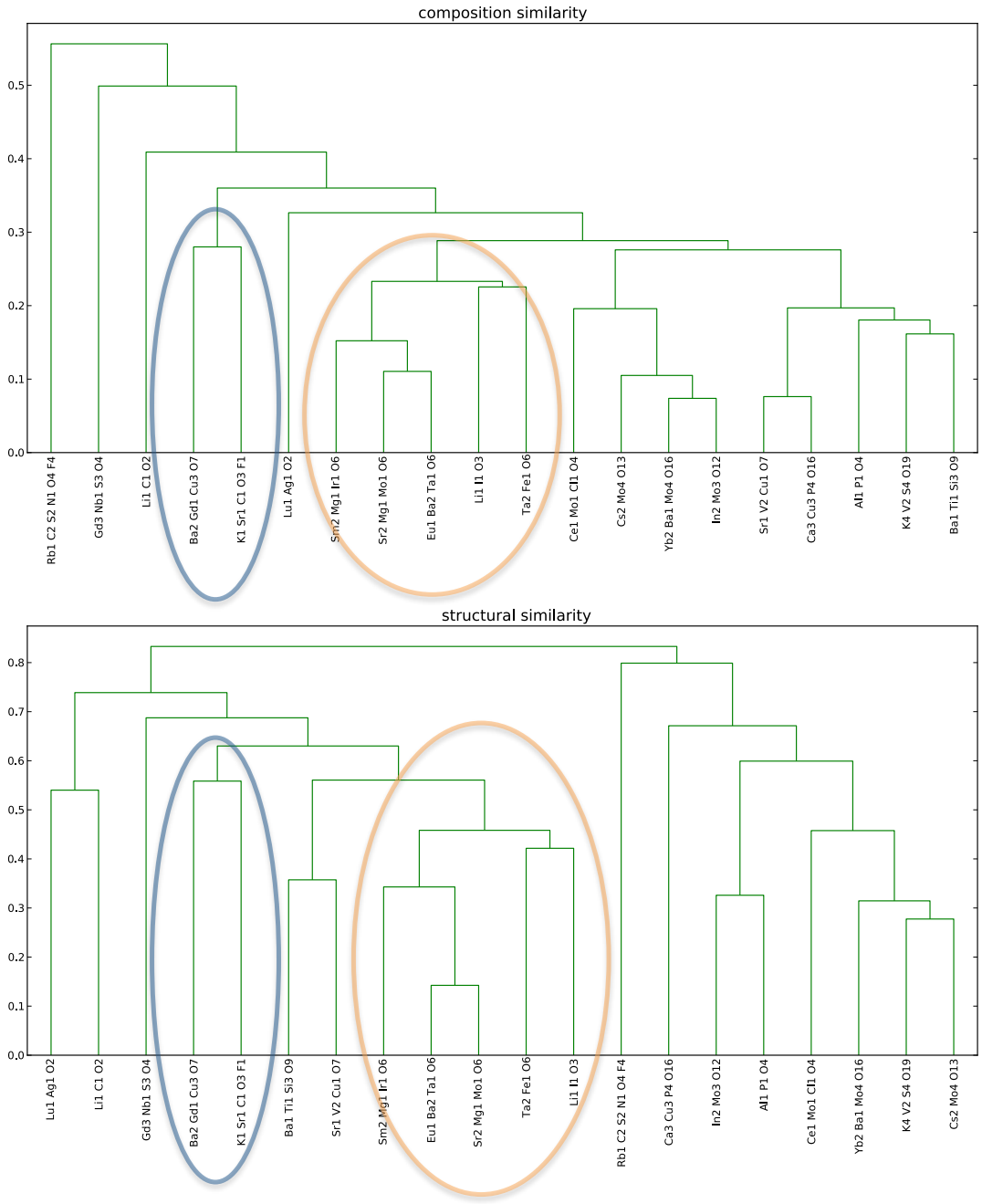


Figure 5.1: Clustering by composition and structural similarity. 25 randomly selected oxides were clustered by both composition and structure similarity. The distances between two oxides are given by the height of the horizontal line connecting them. Composition and structure similarity often preserve clustering characteristics (circled in blue and yellow).

similarity functions are based upon the same ionic similarity function. The clusters circled in blue and in yellow are reproduced by the two different similarities, whereas the rest of the architecture is not reproduced. Organizing compounds by composition similarity yields similar but different results than organizing compounds by structural similarity.

Such organization of data could be carried out on a large scale by the Materials Project[42], enabling users to search through databases including the Inorganic Crystal Structure Database[7] and thousands of *ab initio* computations. An application that enables searching the database by structural similarity to a given compound would allow for substructure-guided search. For example, a researcher studying lithium ion batteries could search for materials with similar substructures to $LiFePO_4$, hoping to find crystal structures with similar substructures, thus maximizing the chances of preserving good Li-ion conductivity. A researcher studying a particular chemistry could search instead by composition similarity to find compounds that are chemically similar, finding new compounds in slightly differing compositions.

Such a large-scale implementation of this organizational scheme would require millions of similarity computations between compositions, substructures, and crystal structures which could be calculated once and stored in the cloud for future use. Ranking a new incoming crystal structure or composition by similarity to every other entry in the database would require tens of thousands of similarity calculations. While similarity calculations are highly parallelizable, this process could be shortened via the implementation of a binary search tree. The implementation of a binary search tree would cut the computational cost of ranking one compound versus n database compounds from n computations down to $2 \log n$ computations.[18]

Unfortunately, one weakness of the similarity algorithms presented in this thesis lies in the fact that the distances derived from them do not satisfy the triangle inequality, and thus the efficacy of a binary search tree would be suspect. If there were one clear improvement to be made to this thesis, it would be to modify the distances presented such that they satisfy the triangle inequality, which would greatly decrease computation time and enable the use of many machine learning techniques for clustering and classification.

At the same time, a brief analysis of the similarity computations made for this thesis show relatively few violations of the triangle inequality; those violations tend to be

small in magnitude. Implementing a binary search tree that confirms the each binary decision by computing the next four options in addition to the current two would be a reasonable method to reduce approximation error. Such a 'double checked' binary search tree would find the most similar structure in $4 \log n$ computations.

5.2.3 Prediction of New Crystal Structures

The work in this thesis also has potential applications to the prediction of new crystal structures. The substructural work in this thesis could be used to predict substructural components of new crystal structures in several ways. However, the packing of substructures in a new crystal structures remains a hard problem, and the work in this thesis would appeal to ongoing research for methods to assist in computing low-energy packings.

For example, given a new composition, we could look for compounds with similar compositions as in chapter 3. These compounds could be used to seed a genetic algorithm such as Glass et al's USPEX[28,52]. The USPEX algorithm could be used to generate mutations between generations of likely crystal structures. The energy of each mutated crystal structure could be evaluated using *ab initio* methods, or as a faster screening process, we could relax each mutated crystal structure to a local minimum using a faster molecular simulation code such as GULP[26], and use not energy, but crystal structural similarity to likely structures to evaluate fitness.

Mellot-Draznieks et al. have published several interesting studies using simulated annealing techniques to assemble "secondary building units."[56,57,58,59] These secondary building units are predefined inorganic building units in three-dimensional spaces. Mellot-Draznieks uses empirical "glueing" rules[58] to scan through potential packings of these building units. Dyer et al. expanded upon this work in 2013, using larger modules to investigate perovskite-related materials.[22,14] While the assembly methods used by Draznieks and Dyer are extraordinarily promising, these structure prediction techniques still depend upon the user's chemical intuition to select plausible secondary building units or modules. The work in this thesis provides a framework for the systematic, quantifiable identification of likely building blocks from existing data. Compounds with similar compositions to the target composition could be decomposed into substructures, and those substructures could be used to seed Mellot-Drazniek's

simulated annealing code.

Additionally, the substructural decomposition in this thesis can be used as a machine learning framework to extract the "glueing" rules between substructures. The decomposition of a crystal structure into substructures described in chapter 4 yields one substructure per ion. Given two substructures a and b with a common ion x, we can now data mine the substructure around ion x. How often are substructures a and b connected by one ion x? How often are they connected by two ions? With what frequency does the substructure around ion x overlap with substructures a and b? Aided by the creation of a large database of substructures, we can answer all of these questions.

5.2.4 Data mining for Materials Properties

Finally, the work in this thesis can be used to search for materials properties. The work in chapter 4 was used to search for Li insertion sites; a straightforward extension of this work could be used to aid in the search for Li diffusion paths.

Rupp et al. have recently presented a similar framework for the prediction of molecular atomization energies.[80] Rupp represents molecules via a Coulomb matrix as described in figure 5.2. The square matrix has one row per atom and can be extracted from the crystal structure of the molecule. The off-diagonal elements of the Coulomb matrix encode the Coulomb repulsion between atoms, whereas the diagonal elements encode a polynomial fit of atomic energies to nuclear charge. Rupp achieves good prediction of atomization energies across 7,000 'small' organic molecules that contain up to 7 heavy atoms, contain C, N, O, or S, and are saturated with hydrogen.

Rupp's algorithm is interesting because it encodes crystal structure information into a two-dimensional matrix that, when passed into a ridge regression algorithm, achieves very good results. However, the weakness of Rupp's algorithm lies in the fact that there is no clear ordering of the atoms in the Coulomb matrix. Although a matrix can only represent one molecule, a molecule can be represented by many matrices by via permutations of the rows and columns of it's Coulomb matrix. Rupp partially circumvents this difficulty by reducing the information encoded in each Coulomb matrix to it's eigenvalue and computing the eigenvalues of many permutations of the Coulomb matrix.[80,31] Given two identical but large and complex molecules, Rupp's algorithm would

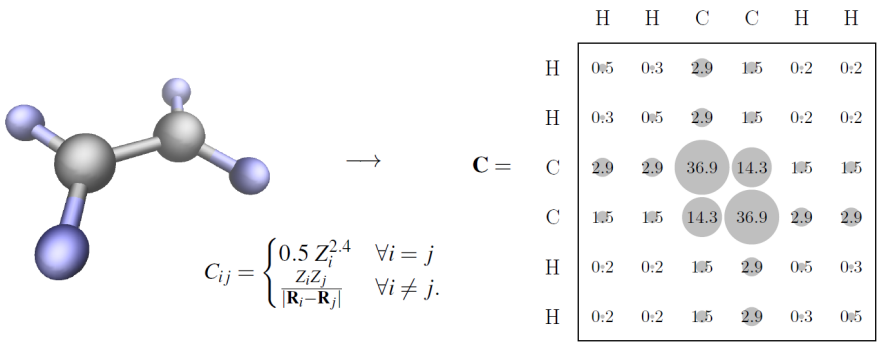


Figure 5.2: Rupp's matrix representation of molecules.[31] The Coulomb matrix representation is extracted from a molecule from the atomic coordinates $\mathbf{R}_i$ and nuclear charges $\mathbf{Z}_i$. The matrix contains one row per atom, is symmetric, and requires no explicit bond information.

need to calculate the eigenvalues of hundreds of permutations of the Coulomb matrix in order to discover that the atomization energies of the two molecules are identical.

The ideal solution to Rupp's problem would involve ordering the atoms each molecular Coulomb matrix in a fashion such that each molecule has only one Coulomb matrix representation. If such an ordering existed, a calculation of the distances between Coulomb matrices would allow for the comparison of the most similar atoms in each molecule. In computer science, the problem of discovering whether or not two Coulomb matrices represent the same molecule is known as the graph isomorphism problem.[18]. It is not currently known if the graph isomorphism problem can be solved in polynomial time.[27,5,4] For large molecules, the graph isomorphism problem is computationally difficult to solve.

The substructural and structural similarity functions in this thesis bypass this difficulty by representing only the Voronoi neighbor interactions. Instead of preserving the entire adjacency matrix, which is computed by Voronoi neighbor adjacency in this thesis, we preserve only the local interactions. This simplification allows us to calculate distances between adjacency matrices by matching the most similar ions of one structure to the most similar ions of the other via the use of a weighted bipartite graph matching algorithm which is calculable in $\mathcal{O}(n^3)$ time.[18] Inspired by Rupp's proof of concept, we believe that the direct, local, geometric and chemical comparisons enabled by this thesis will allow for the data mined prediction of materials properties.

We have presented a structure-based framework for the datamined analysis of crystal

structure. Given a data-mined similarity function between two ions, we have implemented similarity functions between compositions, crystal structures and substructures. These similarity functions can be used to quantitatively organize data, to assist in the prediction of novel crystal structures, and to data mine predictions of materials properties.

References

[1] Abraham, N. & Probert, M. (2006). A periodic genetic algorithm with real-space representation for crystal structure and polymorph prediction. *Physical Review B*, 73(22), 224104–.

[2] Akbarzadeh, A., Wolverton, C., & Ozolins, V. (2009). First-principles determination of crystal structures, phase stability, and reaction thermodynamics in the li-mg-al-h hydrogen storage system. *Physical Review B*, 79(18), 184102.

[3] Aydinol, M. K., Kohan, A. F., Ceder, G., Cho, K., & Joannopoulos, J. (1997). Ab initio study of lithium intercalation in metal oxides and metal dichalcogenides. *Phys. Rev. B*, 56, 1354–1365.

[4] Babai, L. & Codenotti, P. (2008). Isomorhism of hypergraphs of low rank in moderately exponential time. In *Foundations of Computer Science, 2008. IEEE 49th Annual IEEE Symposium on* (pp. 667–676).: IEEE.

[5] Babai, L. & Luks, E. M. (1983). Canonical labeling of graphs. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing* (pp. 171–183).: ACM.

[6] Barber, C. B., Dobkin, D. P., & Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4), 469–483.

[7] Bergerhoff, G., Brown, I., Allen, F., Bergerhoff, G., & Sievers, R. (1987). Crystallographic databases. *Chester: International Union of Crystallography*.

[8] Brown, I. (1981). The bond-valence method: an empirical approach to chemical structure and bonding. *Structure and bonding in crystals*, 2, 1–30.

[9] Brown, I. (1997). Bond valence methods. *Computer Modelling in Inorganic Crystallography*, (pp. 23–53).

[10] Brunner, G. t. & Schwarzenbach, D. (1971). Zur abgrenzung der koordinationssphäre und ermittlung der koordinationszahl in kristallstrukturen*. *Zeitschrift fur Kristallographie*, 133, 127–133.

[11] Burzlaff, H. & Malinovsky, Y. (1997). A Procedure for the Clasification of Non-Organic Crystal Structures. I. Theoretical Background. *Acta Crystallographica Section A*, 53(2), 217–224.

[12] Bush, T. S., Catlow, C. R. A., & Battle, P. D. (1995). Evolutionary programming techniques for predicting inorganic crystal structures. *Journal of Materials Chemistry*, 5(8), 1269.

[13] Carter, E. A. (2008). Challenges in modeling materials properties without experimental input. *Science*, 321(5890), 800–803.

[14] Catlow, R. (2013). Inorganic materials: Intuition weaved into computation. *Nat Chem*, 5(8), 648–649.

[15] Ceder, G., Aydinol, M., & Kohan, A. (1997). Application of first-principles calculations to the design of rechargeable li-batteries. *Computational Materials Science*, 8(1–2), 161 – 169. Proceedings of the joint NSF/CNRS Workshop on Alloy Theory.

[16] Clapper, B. (2013). Munkres. http://software.clapper.org/munkres/.

[17] Coley, D. A. (1999). *An introduction to genetic algorithms for scientists and engineers*. World scientific.

[18] Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., et al. (2001). *Introduction to algorithms*, volume 2. MIT Press, Cambridge.

[19] Curtarolo, S., Morgan, D., Persson, K., Rodgers, J., & Ceder, G. (2003). Predicting crystal structures with data mining of quantum calculations. *Physical review letters*, 91(13), 135503.

[20] Daams, J., Van Vucht, J., & Villars, P. (1992). Atomic-environment classification of the cubic "intermetallic" structure types. *Journal of Alloys and Compounds*, 182(1), 1–33.

[21] Daams, J. & Villars, P. (2000). Atomic environments in relation to compound prediction. *Engineering Applications of Artificial Intelligence*, 13(5), 507 – 511.

[22] Dyer, M. S., Collins, C., Hodgeman, D., Chater, P. A., Demont, A., Romani, S., Sayers, R., Thomas, M. F., Claridge, J. B., Darling, G. R., & Rosseinsky, M. J. (2013). Computationally assisted identification of functional inorganic materials. *Science*, 340(6134), 847–852.

[23] Fischer, C. C. (2007). *A Machine Learning Approach to Crystal Structure Prediction*. PhD thesis, Massachusetts Institute of Technology.

[24] Fischer, C. C., Tibbetts, K. J., Morgan, D., & Ceder, G. (2006). Predicting crystal structure by merging data mining with quantum mechanics. *Nature materials*, 5(8), 641–646.

[25] Frank, F. t. & Kasper, J. (1958). Complex alloy structures regarded as sphere packings. i. definitions and basic principles. *Acta Crystallographica*, 11(3), 184–190.

[26] Gale, J. (1992). Gulp (general utility lattice program). *Royal Institution, London*.

[27] Garey, M. R. & Johnson, D. S. (1979). Computers and intractability: a guide to np-completeness.

[28] Glass, C. W., Oganov, A. R., & Hansen, N. (2006). USPEX—Evolutionary crystal structure prediction. *Computer Physics Communications*, 175(11-12), 713–720.

[29] Greeley, J. & Mavrikakis, M. (2004). Alloy catalysts designed from first principles. *Nature materials*, 3(11), 810–815.

[30] Gross, E. K. U. & Dreizler, R. M. (1995). *Density functional theory*, volume 337. Springer.

[31] Hansen, K., Montavon, G., Biegler, F., Fazli, S., Rupp, M., Scheffler, M., von Lilienfeld, O. A., Tkatchenko, A., & Müller, K.-R. (2013). Assessment and validation of machine learning methods for predicting molecular atomization energies. *Journal of Chemical Theory and Computation*, 9(8), 3404–3419.

[32] Harrison, W. A. (1966). *Pseudopotentials in the Theory of Metals*, volume 201. WA Benjamin New York.

[33] Hartke, B. (2004). Application of evolutionary algorithms to global cluster geometry optimization. In *Applications of Evolutionary Computation in Chemistry* (pp. 33–53). Springer.

[34] Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., & Tibshirani, R. (2009). *The elements of statistical learning*, volume 2. Springer.

[35] Hautier, G. (2011). *High-Throughput Data Mined Prediction of Inorganic Compounds and Computational Discovery of New Lithium-Ion Battery Cathode Materials*. PhD thesis, Massachusetts Institute of Technology.

[36] Hautier, G., Fischer, C., Ehrlacher, V., Jain, A., & Ceder, G. (2011). Data Mined Ionic Substitutions for the Discovery of New Compounds. *Inorganic Chemistry*, 50(2), 656–663.

[37] Hautier, G., Fischer, C. C., Jain, A., Mueller, T., & Ceder, G. (2010). Finding Nature's Missing Ternary Oxide Compounds Using Machine Learning and Density Functional Theory. *Chemistry of Materials*, 22(12), 3762–3767.

[38] Heine, V. & Weaire, D. (1966). Structure of di-and trivalent metals. *Physical Review*, 152(2), 603.

[39] Hume-Rothery, W., Smallman, R. E., & Haworth, C. W. (1969). *The structure of metals and alloys*, volume 1. Metals & Metallurgy Trust.

[40] Hundt, R., Schön, J. C., & Jansen, M. (2006). CMPZ - an algorithm for the efficient comparison of periodic structures. *Journal of Applied Crystallography*, 39(1), 6–16.

[41] Jain, A., Hautier, G., Moore, C. J., Ong, S. P., Fischer, C. C., Mueller, T., Persson, K. A., & Ceder, G. (2011). A high-throughput infrastructure for density functional theory calculations. *Computational Materials Science*, 50(8), 2295–2310.

[42] Jain, A., Ong, S. P., Hautier, G., Chen, W., Richards, W. D., Dacek, S., Cholia, S., Gunter, D., Skinner, D., Ceder, G., & Persson, K. a. (2013). The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1), 011002.

[43] John, J. & Bloch, A. N. (1974). Quantum-defect electronegativity scale for nontransition elements. *Physical Review Letters*, 33, 1095–1098.

[44] Johnston, R. L. (2003). Evolving better nanoparticles: Genetic algorithms for optimising cluster geometries. *Dalton Transactions*, (22), 4193–4207.

[45] Jones, E., Oliphant, T., Peterson, P., et al. (2001–). SciPy: Open source scientific tools for Python. http://www.scipy.org/.

[46] Kang, S., Mo, Y., Ong, S. P., & Ceder, G. (2013). A facile mechanism for recharging Li_2O_2 in $Li - O_2$batteries. *Chemistry of Materials*, 25(16), 3328–3336.

[47] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983a). Optimization by simulated annealing. *Science (New York, N.Y.)*, 220(4598), 671–80.

[48] Kirkpatrick, S., Jr., D. G., & Vecchi, M. P. (1983b). Optimization by simmulated annealing. *science*, 220(4598), 671–680.

[49] Le Bail, A. (2005). Inorganic structure prediction with GRINSP. *Journal of Applied Crystallography*, 38(2), 389–395.

[50] Lloyd, L. D., Johnston, R. L., & Salhi, S. (2005). Strategies for increasing the efficiency of a genetic algorithm for the structural optimization of nanoalloy clusters. *Journal of computational chemistry*, 26(10), 1069–1078.

[51] Lopez-Bezanilla, A. & von Lilienfeld, O. A. (2014). Modeling electronic quantum transport with machine learning. *arXiv preprint, arXiv:1401.8277*.

[52] Lyakhov, A. O., Oganov, A. R., Stokes, H. T., & Zhu, Q. (2013). New developments in evolutionary structure prediction algorithm uspex. *Computer Physics Communications*, 184(4), 1172–1182.

[53] Maddox, J. (1988). Crystals from first principles. *Nature*, 335(6187).

[54] Martin, R. M. (2004). *Electronic Structure: Basic Theory and Practical Methods*. Cambridge University Press.

[55] Marzari, N. (1996). *Ab-initio molecular dynamics for metallic systems*. PhD thesis, University of Cambridge.

[56] Mellot-Draznieks, C., Dutour, J., & Férey, G. (2004). Hybrid organic–inorganic frameworks: Routes for computational design and structure prediction. *Angewandte Chemie International Edition*, 43(46), 6290–6296.

[57] Mellot-Draznieks, C., Girard, S., & Férey, G. (2002a). Novel inorganic frameworks constructed from double-four-ring (d4r) units: computational design, structures, and lattice energies of silicate, aluminophosphate, and gallophosphate candidates. *Journal of the American Chemical Society*, 124(51), 15326–15335.

[58] Mellot-Draznieks, C., Girard, S., Férey, G., Schön, J. C., Cancarevic, Z., & Jansen, M. (2002b). Computational design and prediction of interesting not-yet-synthesized structures of inorganic materials by using building unit concepts. *Chemistry-A European Journal*, 8(18), 4102–4113.

[59] Mellot Draznieks, C., Newsam, J. M., Gorman, A. M., Freeman, C. M., & Férey, G. (2000). De novo prediction of inorganic structures developed through automated assembly of secondary building units (aasbu method). *Angewandte Chemie International Edition*, 39(13), 2270–2275.

[60] Meng, Y. S. & Arroyo-de Dompablo, M. E. (2009). First principles computational materials design for energy storage materials in lithium ion batteries. *Energy Environ. Sci.*, 2(6), 589–609.

[61] Meng, Y. S. & Arroyo-de Dompablo, M. E. (2013). Recent advances in first principles computational research of cathode materials for lithium-ion batteries. *Accounts of Chemical Research*, 46(5), 1171–1180.

[62] Miedema, A. R., de Châtel, P. F., & de Boer, F. R. (1980). Cohesion in alloys — fundamentals of a semi-empirical model. *Physica B+C*, 100(1), 1–28.

[63] Mo, Y., Ong, S. P., & Ceder, G. (2011). First principles study of the li10gep2s12 lithium super ionic conductor material. *Chemistry of Materials*, 24(1), 15–17.

[64] Montavon, G., Hansen, K., Fazli, S., Rupp, M., Biegler, F., Ziehe, A., Tkatchenko, A., von Lilienfeld, A., & Müller, K.-R. (2012). Learning invariant representations of molecules for atomization energy prediction. In *NIPS* (pp. 449–457).

[65] Montavon, G., Rupp, M., Gobre, V., Vazquez-Mayagoitia, A., Hansen, K., Tkatchenko, A., Müller, K.-R., & von Lilienfeld, O. A. (2013). Machine learning of molecular electronic properties in chemical compound space. *New Journal of Physics*, 15(9), 095003.

[66] Mooser, E. & Pearson, W. (1959). On the crystal chemistry of normal valence compounds. *Acta crystallographica*, 12(12), 1015–1022.

[67] Morgan, D., Rodgers, J., & Ceder, G. (2003). Automatic construction, implementation and assessment of pettifor maps. *Journal of Physics: Condensed Matter*, 15(25), 4361.

[68] Muller, O. & Roy, R. (1974). *The major ternary structural families*. Springer New York.

[69] Oganov, A. R., Chen, J., Gatti, C., Ma, Y., Ma, Y., Glass, C. W., Liu, Z., Yu, T., Kurakevych, O. O., & Solozhenko, V. L. (2009). Ionic high-pressure form of elemental boron. *Nature*, 457(7231), 863–867.

[70] Oganov, A. R. & Valle, M. (2009). How to quantify energy landscapes of solids. *The Journal of chemical physics*, 130(10), 104504.

[71] O'Keeffe, M. (1979). A proposed rigorous definition of coordination number. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 35(5), 772–775.

[72] O'Keeffe, M. (2010). Aspects of crystal structure prediction: some successes and some difficulties. *Physical Chemistry Chemical Physics*, 12(30), 8580–8583.

[73] Ong, S. P., Richards, W. D., Jain, A., Hautier, G., Kocher, M., Cholia, S., Gunter, D., Chevrier, V. L., Persson, K. A., & Ceder, G. (2013). Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68, 314–319.

[74] Pannetier, J., Bassas-Alsina, J., Rodriguez-Carvajal, J., & Caignaert, V. (1990). Prediction of crystal structures from crystal chemistry rules by simulated annealing. *Nature*, 346(6282), 343–345.

[75] Pauling, L. (1929). The principles determining the structure of complex ionic crystals. *Journal of the American Chemical Society*, 51(4), 1010–1026.

[76] Pauling, L. (1960). *The nature of the chemical bond and the structure of molecules and crystals: an introduction to modern structural chemistry*, volume 18. Cornell University Press.

[77] Pettifor, D. (2003). Structure maps revisited. *Journal of Physics: Condensed Matter*, 15(25), V13–V16.

[78] Pettifor, D. G. (1986). The structures of binary compounds. I. Phenomenological structure maps. *Journal of Physics C: Solid State Physics*, 19(3), 285.

[79] Ping Ong, S., Wang, L., Kang, B., & Ceder, G. (2008). Li- fe- p- 02 phase diagram from first principles calculations. *Chemistry of Materials*, 20(5), 1798–1807.

[80] Rupp, M., Tkatchenko, A., Müller, K.-R., & von Lilienfeld, O. A. (2012). Fast and accurate modeling of molecular atomization energies with machine learning. *Physical review letters*, 108(5), 058301.

[81] Schön, J. & Jansen, M. (2001). Determination, prediction, and understanding of structures, using the energy landscapes of chemical systems-part i. *Zeitschrift für Kristallographie*, 216(6), 307–325.

[82] Schön, J. C., Doll, K., & Jansen, M. (2010). Predicting solid compounds via global exploration of the energy landscape of solids on the ab initio level without recourse to experimental information. *Physica Status Solidi (b)*, 247(1), 23–39.

[83] Schön, J. C. & Jansen, M. (1996). First step towards planning of syntheses in solid-state chemistry: Determination of promising structure candidates by global optimization. *Angewandte Chemie International Edition in English*, 35(12), 1286–1304.

[84] Simons, G. & Bloch, A. N. (1973). Pauli-force model potential for solids. *Physical Review B*, 7(6), 2754.

[85] Togo, A. (2009). spglib: A c library for finding and handling crystal symmetries. http://spglib.sourceforge.net/.

[86] Van der Ven, A. & Ceder, G. (2000). Lithium diffusion in layered li x coo2. *Electrochemical and Solid-State Letters*, 3(7), 301–304.

[87] Villars, P. (1983). A three-dimensional structural stability diagram for 998 binary ab intermetallic compounds. *Journal of the Less Common Metals*, 92(2), 215–238.

[88] Villars, P. (1986). A semiempirical approach to the prediction of compound formation for 96446 ternary alloy systems: Ii. *Journal of the Less Common Metals*, 119(1), 175–188.

[89] Villars, P. (2000). Factors governing crystal structures. *Intermetallic Compounds: Principles and Practice*, 1, 228.

[90] Villars, P., Berndt, M., Brandenburg, K., Cenzual, K., Daams, J., Hulliger, F., Massalski, T., Okamoto, H., Osaki, K., Prince, A., et al. (2004). The pauling file. *Journal of alloys and compounds*, 367(1), 293–297.

[91] Voronoi, G. (1908). Nouvelles applications des paramètres continus àla théorie des formes quadratiques. premier mémoire. sur quelques propriétés des formes quadratiques positives parfaites. *Journal für die reine und angewandte Mathematik*, 1908, 97.

[92] Wales, D. J. & Doye, J. P. (1997). Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, 101(28), 5111–5116.

[93] Wales, D. J. & Scheraga, H. A. (1999). Global optimization of clusters, crystals, and biomolecules. *Science*, 285(5432), 1368–1372.

[94] Wang, Y., Lv, J., Zhu, L., & Ma, Y. (2010). Crystal structure prediction via particle-swarm optimization. *Phys. Rev. B*, 82(9), 94116.

[95] Wang, Y., Lv, J., Zhu, L., & Ma, Y. (2012). CALYPSO: A method for crystal structure prediction. *Computer Physics Communications*, 183(10), 2063–2070.

[96] Woodley, S. M. (2007). Engineering microporous architectures: combining evolutionary algorithms with predefined exclusion zones. *Physical Chemistry Chemical Physics*, 9(9), 1070–1077.

[97] Woodley, S. M., Battle, P. D., Gale, J. D., & Catlow, C. R. A. (2003). Computer-Simulation Study of the Orthorhombic–Hexagonal Phase Change in Lanthanide Manganates ($LnMnO_3$). *Chemistry of Materials*, 15(8), 1669–1675.

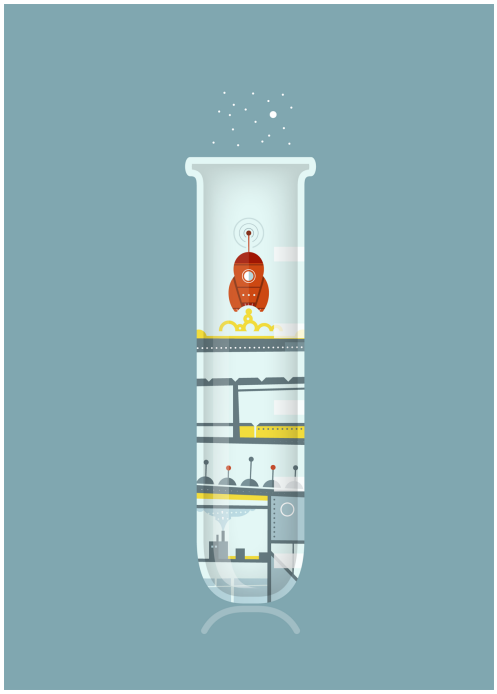
[98] Woodley, S. M. & Catlow, R. (2008). Crystal structure prediction from first principles. *Nat Mater*, 7(12), 937–946.

[99] Wu, Y. & Ceder, G. (2013). First principles study on ta3n5: Ti3o3n2 solid solution as a water-splitting photocatalyst. *The Journal of Physical Chemistry C*, 117(47), 24710–24715.

[100] Zhou, F., Cococcioni, M., Kang, K., & Ceder, G. (2004). The li intercalation potential of $LiMPO_4$ and $LiMSiO_4$ olivines with M= Fe, Mn, Co, Ni. *Electrochemistry communications*, 6(11), 1144–1148.

[101] Zhu, Q., Oganov, A. R., Glass, C. W., & Stokes, H. T. (2012a). Constrained evolutionary algorithm for structure prediction of molecular crystals: methodology and applications. *Acta Crystallographica Section B: Structural Science*, 68(3), 215–226.

[102] Zhu, Q., Oganov, A. R., & Lyakhov, A. O. (2012b). Evolutionary metadynamics: a novel method to predict crystal structures. *CrystEngComm*, 14(10), 3596–3601.



This thesis was typeset using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. The body text is set in 11 point Egenolff-Berner Garamond, a revival of Claude Garamont's humanist typeface. The above illustration, "Science Experiment 02", was created by Ben Schlitter and released under CC BY-NC-ND 3.0. A template that can be used to format a PhD thesis with this look and feel has been released under the permissive MIT (XII) license, and can be found online at github.com/suchow/Dissertate or from its author, Jordan Suchow, at suchow@post.harvard.edu.